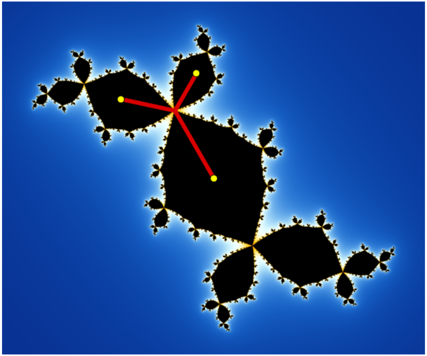
Recognizing Topological Polynomials by Lifting Trees



Jim Belk, University of St Andrews

Collaborators

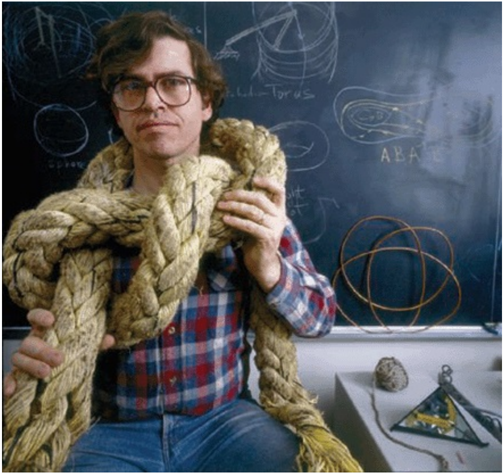
Justin Lanier,
Georgia Tech

Dan Margalit,
Georgia Tech

Becca Winarski,
U. Michigan

Topological Polynomials

In the 1980's, Bill Thurston began to study complex polynomials from a topological viewpoint.



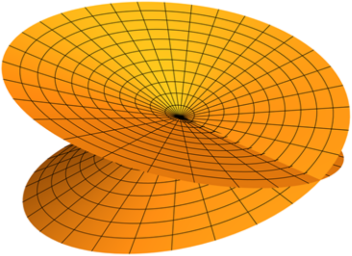
Topological Polynomials

In the 1980's, Bill Thurston began to study complex polynomials from a topological viewpoint.

A ***topological polynomial*** is any orientation-preserving branched cover

$$f : \mathbb{C} \to \mathbb{C}$$

with finitely many branch points.

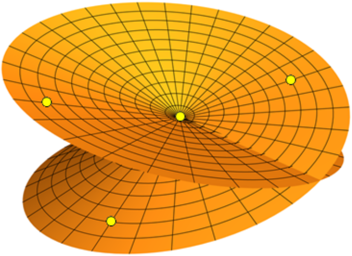


In analogy with polynomials, we refer to the branch points as ***critical points***, and their images as ***critical values***.

Marked Points

We can ***mark*** a topological polynomial by choosing a finite set $M \subset \mathbb{C}$, where

1. $f(M) \subset M$, and
2. M contains the critical values of f.



Marked Points

We can ***mark*** a topological polynomial by choosing a finite set $M \subset \mathbb{C}$, where

1. $f(M) \subset M$, and
2. M contains the critical values of f.

The pair (f, M) is a ***marked topological polynomial***.

Marked Points

We can ***mark*** a topological polynomial by choosing a finite set $M \subset \mathbb{C}$, where

1. $f(M) \subset M$, and
2. M contains the critical values of f.

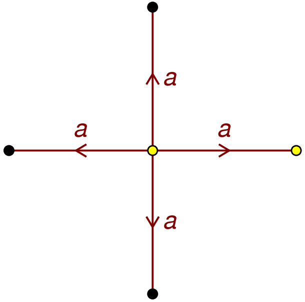
The pair (f, M) is a ***marked topological polynomial***.

Thurston's Question: Is (f, M) "topologically equivalent" to a polynomial?

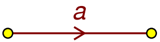
Examples

We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.



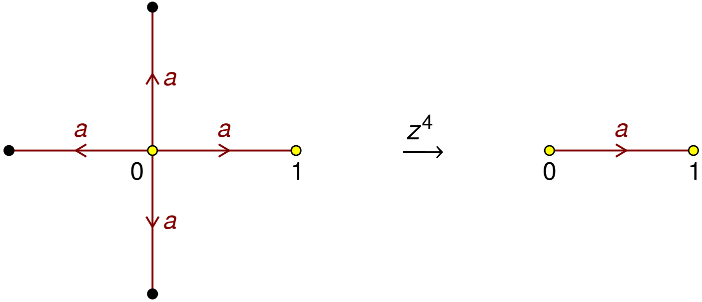
$\longrightarrow$



Examples

We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.



Examples

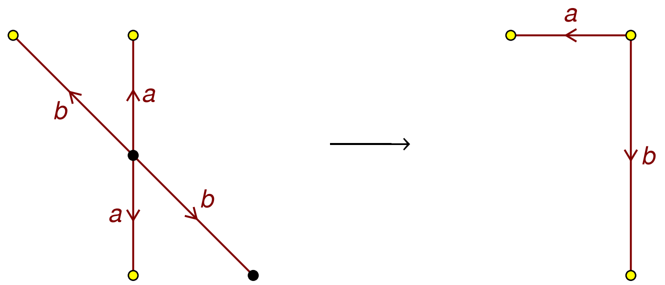
We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.

Examples

We can specify (f, M) up to isotopy by drawing

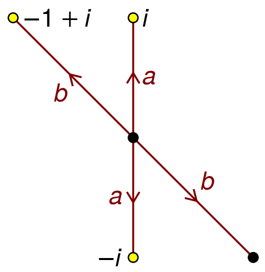
1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.



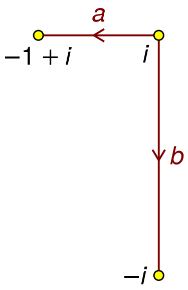
Examples

We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.



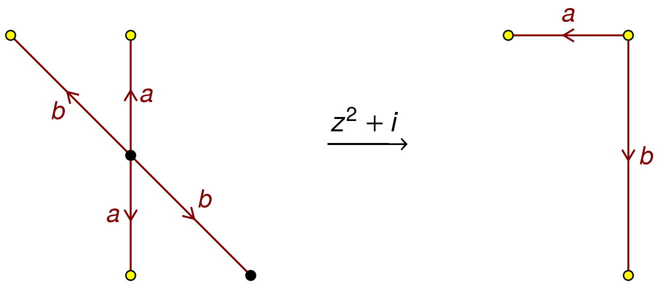
$\xrightarrow{z^2 + i}$



Examples

We can specify (f, M) up to isotopy by drawing

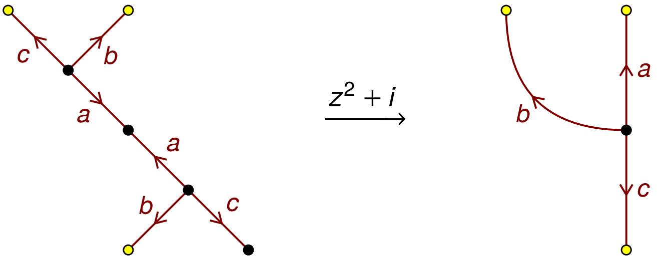
1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.



Examples

We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.



Examples

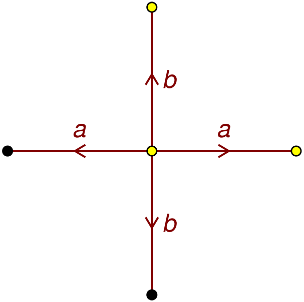
We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.

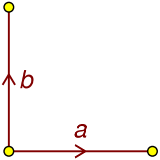
Examples

We can specify (f, M) up to isotopy by drawing

1. Any tree T containing M,
2. Its preimage $f^{-1}(T)$, and the mapping $f^{-1}(T) \to T$.

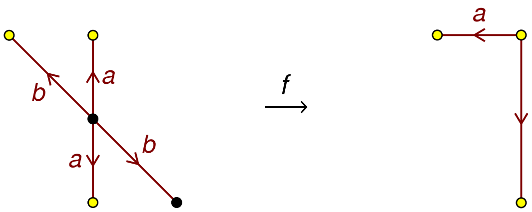


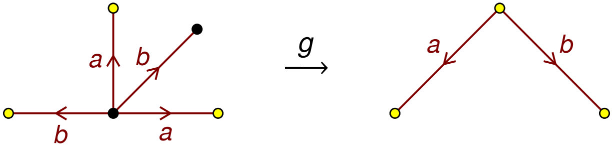
$$\longrightarrow$$



Definition of Topological Equivalence

Two marked topological polynomials are ***topologically equivalent*** if there is a homeomorphism conjugating one to the other.





Definition of Topological Equivalence

Two marked topological polynomials are ***topologically equivalent*** if there is a homeomorphism conjugating one to the other.

$$\begin{tikzcd}
(\mathbb{C}, M) \arrow[r, "f"] \arrow[d, "h"'] & (\mathbb{C}, M) \arrow[d, "h"'] \\
(\mathbb{C}, N) \arrow[r, "g"] & (\mathbb{C}, N)
\end{tikzcd}$$

Thurston’s Theorems

Thurston's Theorems

Thurston Rigidity (1982)

Suppose that (f, M) and (g, N) are topologically equivalent:

$$\begin{tikzcd}
(\mathbb{C}, M) \arrow[r, "f"] \arrow[d, "h"'] & (\mathbb{C}, M) \arrow[d, "h"] \\
(\mathbb{C}, N) \arrow[r, "g"'] & (\mathbb{C}, N)
\end{tikzcd}$$

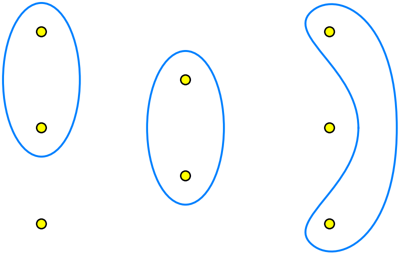
If f and g are complex polynomials, then h has the form

$$h(z) = az + b.$$

Thurston's Theorems

Thurston's Theorem (1982)

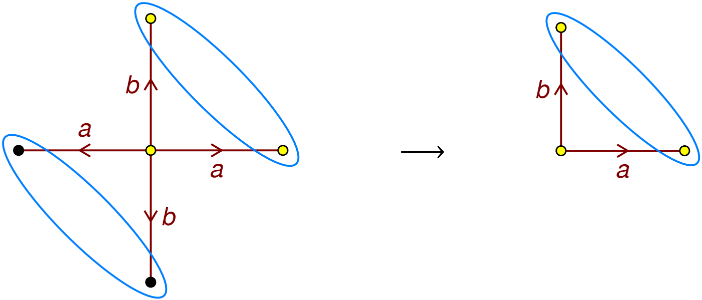
A marked polynomial (f, M) is topologically equivalent to a complex polynomial if and only if it has no ***Thurston obstruction***.



Thurston's Theorems

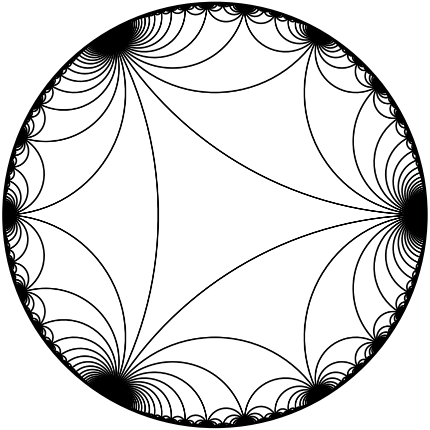
Thurston's Theorem (1982)

A marked polynomial (f, M) is topologically equivalent to a complex polynomial if and only if it has no ***Thurston obstruction***.



Thurston's Theorems

Thurston proved both of these theorems by analyzing the dynamics of the ***pullback map*** on Teichmüller space.



Algorithmic Question

So given a marked topological polynomial (f, M), exactly one of the following holds:

1. (f, M) is topologically equivalent to a complex polynomial, unique up to affine conjugacy.

2. (f, M) has a Thurston obstruction.

Algorithmic Question

So given a marked topological polynomial (f, M), exactly one of the following holds:

1. (f, M) is topologically equivalent to a complex polynomial, unique up to affine conjugacy.
2. (f, M) has a Thurston obstruction.

Question: How can we distinguish between these two cases? How do we actually find the complex polynomial (in case 1) or Thurston obstruction (in case 2)?

The Twisted Rabbit Problem

Hubbard (1983) observed that this is difficult even for $\deg(f) = 2$ and $|M| = 3$. This is the ***twisted rabbit problem***.

John Hubbard

The Twisted Rabbit Problem

Hubbard (1983) observed that this is difficult even for $\deg(f) = 2$ and $|M| = 3$. This is the ***twisted rabbit problem***.

This case was solved by Bartholdi and Nekrashevych in 2006 using ***iterated monodromy groups***.

Laurent Bartholdi

Volodymyr Nekrashevych

The Twisted Rabbit Problem

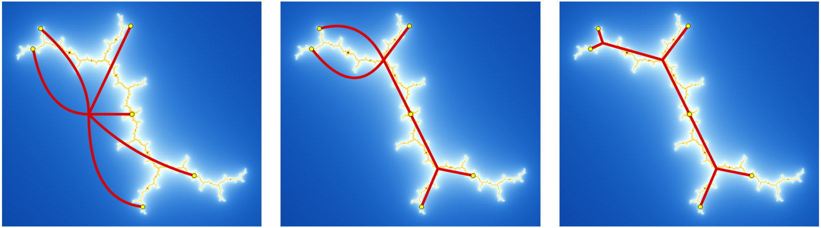
Hubbard (1983) observed that this is difficult even for $\deg(f) = 2$ and $|M| = 3$. This is the ***twisted rabbit problem***.

This case was solved by Bartholdi and Nekrashevych in 2006 using ***iterated monodromy groups***.

Unfortunately, Bartholdi and Nekrashevych's methods are difficult to apply for $\deg(f) \geq 3$ or $|M| \geq 4$.

Main Result

We have developed a simple geometric algorithm (the ***tree lifting algorithm***) that answers these questions much more generally.



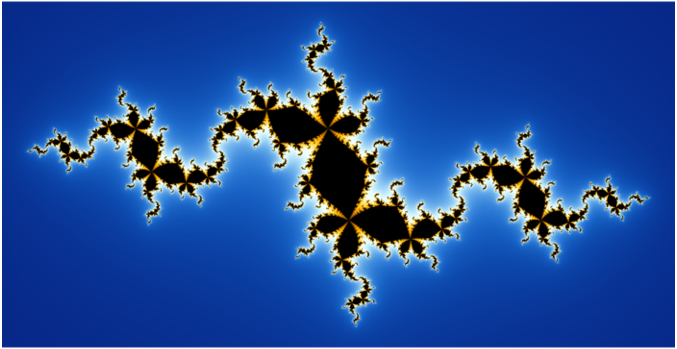
Given an (f, M), the algorithm produces either

1. The Hubbard tree for a polynomial equivalent to (f, M), or
2. The canonical Thurston obstruction for (f, M).

Some Complex Dynamics

Julia Sets

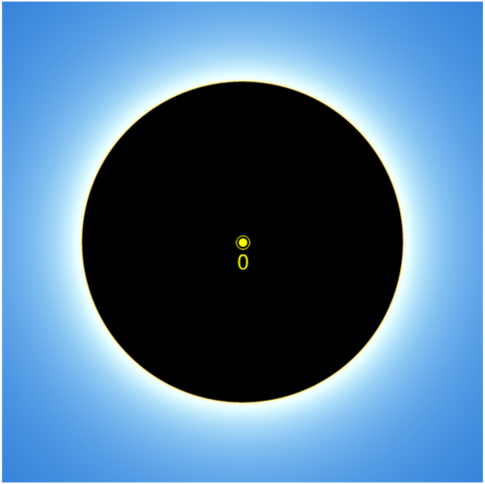
Every complex polynomial $f(z)$ of degree ≥ 2 has a ***filled Julia set***.



This is the unique maximal compact, f-invariant subset of $\mathbb{C}$.

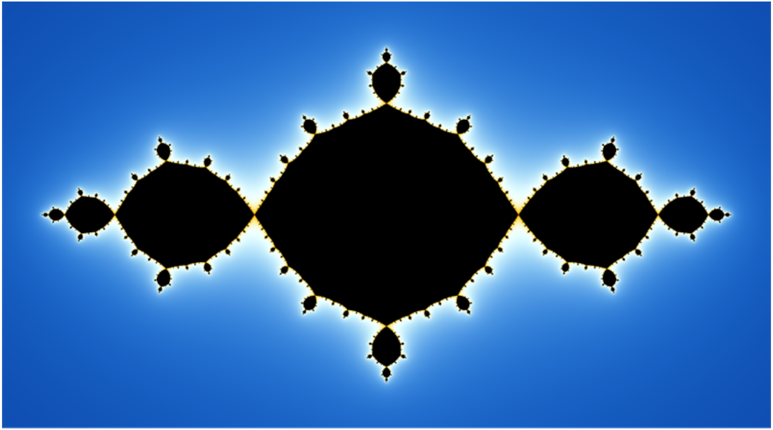
Julia Sets

For example, the filled Julia set for $f(z) = z^2$ is the closed unit disk.



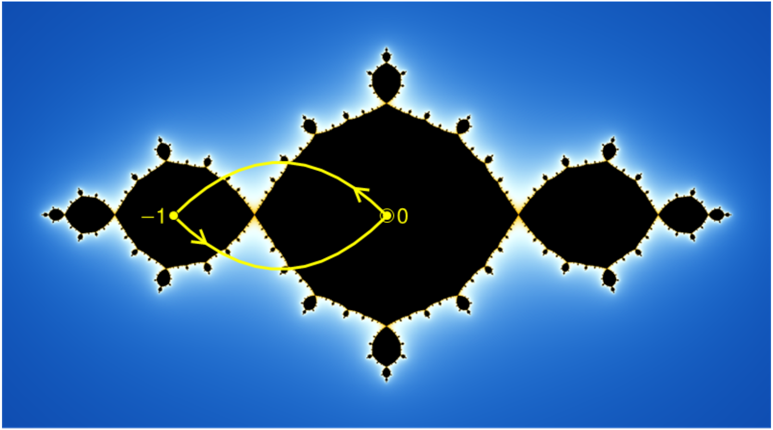
Julia Sets

Here's the filled Julia set for $z^2 - 1$.

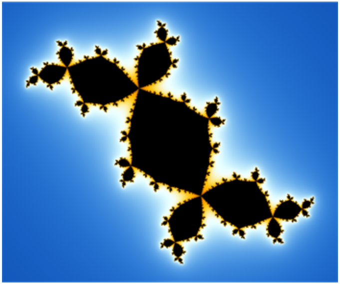


Julia Sets

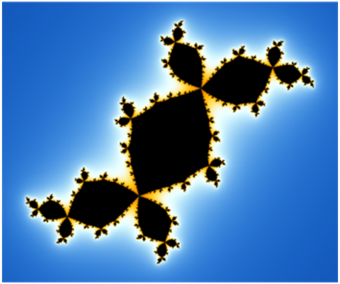
Here's the filled Julia set for $z^2 - 1$.



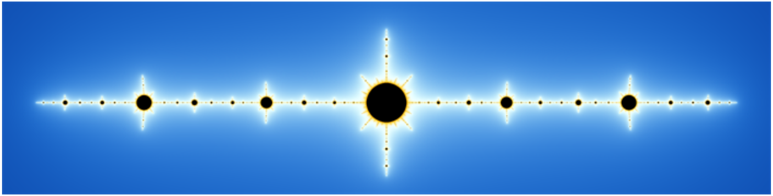
Julia Sets



rabbit

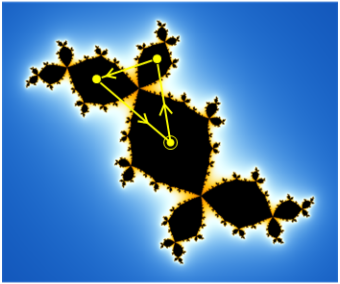


corabbit

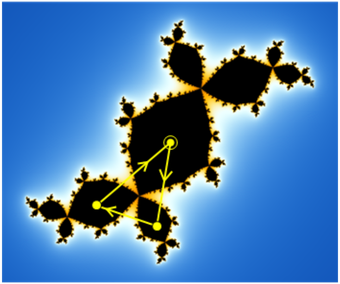


airplane

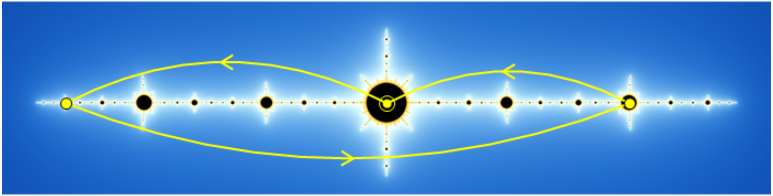
Julia Sets



rabbit



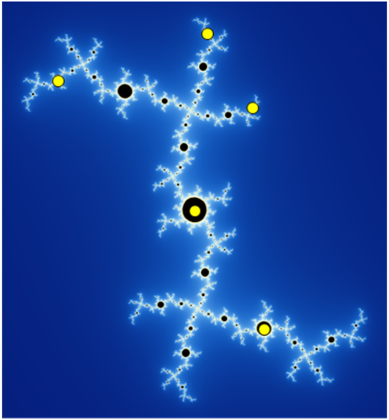
corabbit



airplane

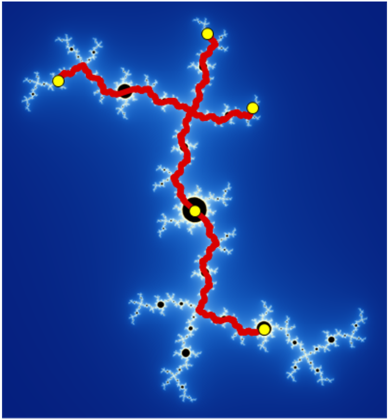
The Hubbard Tree

We can use the Julia set to define the ***Hubbard tree*** H, a special tree that contains M.



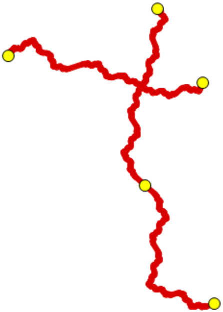
The Hubbard Tree

We can use the Julia set to define the ***Hubbard tree*** H, a special tree that contains M.



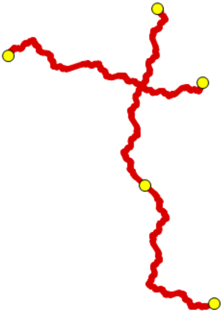
The Hubbard Tree

We can use the Julia set to define the ***Hubbard tree*** H, a special tree that contains M.



The Hubbard Tree

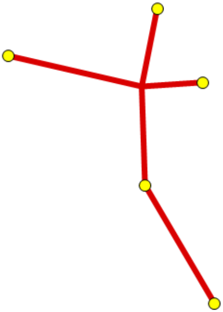
We can use the Julia set to define the ***Hubbard tree*** H, a special tree that contains M.



The Hubbard tree is forward invariant, i.e. $f(H) \subseteq H$.

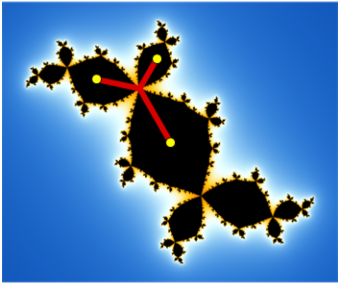
The Hubbard Tree

We can use the Julia set to define the ***Hubbard tree*** H, a special tree that contains M.

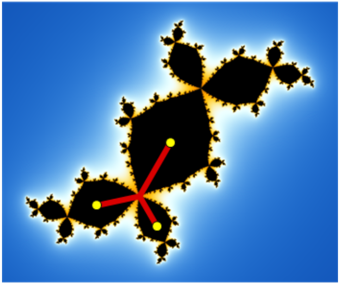


The Hubbard tree is forward invariant, i.e. $f(H) \subseteq H$.

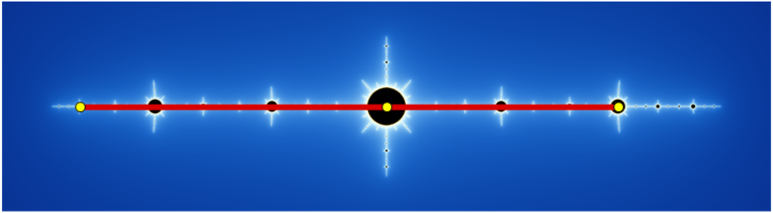
The Hubbard Tree



rabbit



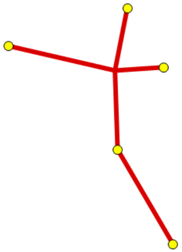
corabbit



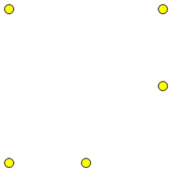
airplane

The Hubbard Tree

Any map that's topologically equivalent to a polynomial has a ***topological Hubbard tree***.

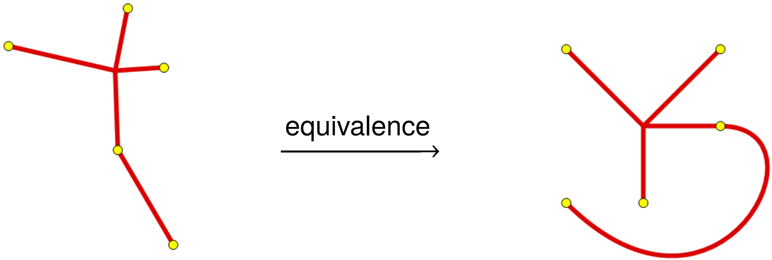


$\xrightarrow{\text{equivalence}}$



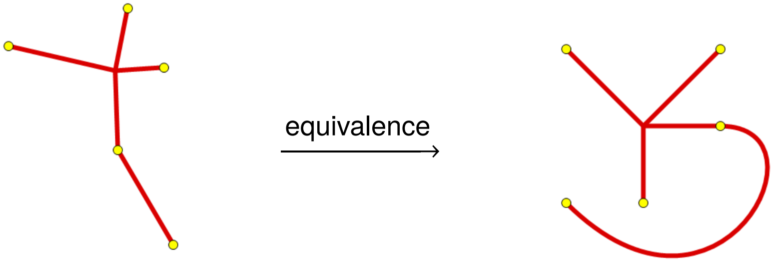
The Hubbard Tree

Any map that's topologically equivalent to a polynomial has a ***topological Hubbard tree***.



The Hubbard Tree

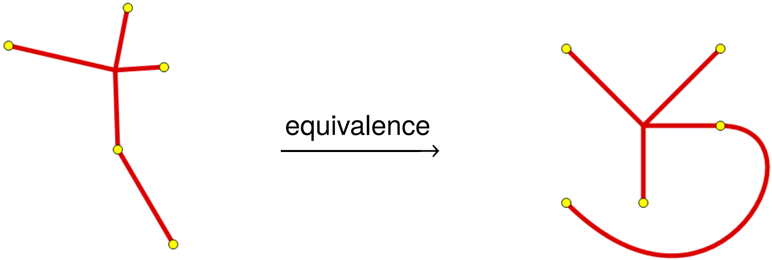
Any map that's topologically equivalent to a polynomial has a ***topological Hubbard tree***.



Idea: Use topology to find the topological Hubbard tree.

The Hubbard Tree

Any map that's topologically equivalent to a polynomial has a ***topological Hubbard tree***.



Idea: Use topology to find the topological Hubbard tree.

Note: The polynomial $f(z)$ is determined by H and the mapping

$$f^{-1}(H) \longrightarrow H.$$

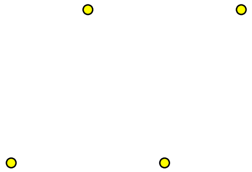
Lifting Trees

Allowed Trees

We will consider trees in $\mathbb{C}$ that satisfy the following conditions:

1. T contains M, and
2. Every leaf of T lies in M.

Isotopic trees are considered the same.

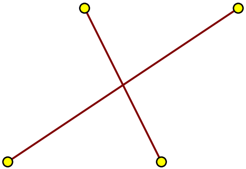


Allowed Trees

We will consider trees in $\mathbb{C}$ that satisfy the following conditions:

1. T contains M, and
2. Every leaf of T lies in M.

Isotopic trees are considered the same.

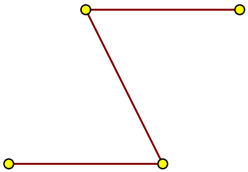


Allowed Trees

We will consider trees in $\mathbb{C}$ that satisfy the following conditions:

1. T contains M, and
2. Every leaf of T lies in M.

Isotopic trees are considered the same.

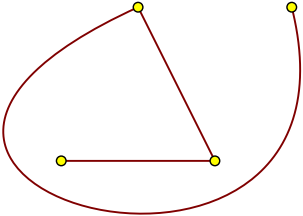


Allowed Trees

We will consider trees in $\mathbb{C}$ that satisfy the following conditions:

1. T contains M, and
2. Every leaf of T lies in M.

Isotopic trees are considered the same.

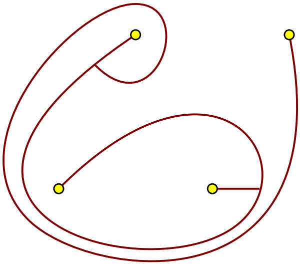


Allowed Trees

We will consider trees in $\mathbb{C}$ that satisfy the following conditions:

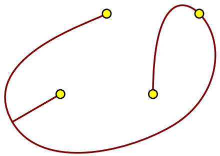
1. T contains M, and
2. Every leaf of T lies in M.

Isotopic trees are considered the same.

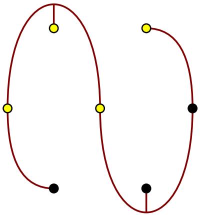


Lifting Trees

The preimage $f^{-1}(T)$ of an allowed tree is not an allowed tree.



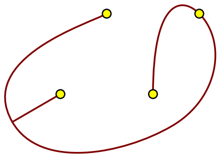
Tree T



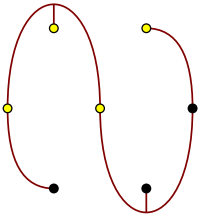
preimage $f^{-1}(T)$

Lifting Trees

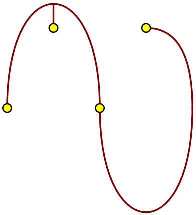
The preimage $f^{-1}(T)$ of an allowed tree is not an allowed tree.



Tree T



preimage $f^{-1}(T)$



Lift $\lambda_f(T)$

The ***lift*** of T is the subtree of $f^{-1}(T)$ spanned by M.

Lifting Trees

For a given M, lifting under f defines a function

$$\lambda_f : \text{allowed trees} \longrightarrow \text{allowed trees}$$

Lifting Trees

For a given M, lifting under f defines a function

$$\lambda_f : \text{allowed trees} \longrightarrow \text{allowed trees}$$

Fact: The (topological) Hubbard tree is a fixed point for λ_f.

This is because the Hubbard tree H satisfies

$$H \subset f^{-1}(H).$$

Lifting Trees

For a given M, lifting under f defines a function

$$\lambda_f : \text{allowed trees} \longrightarrow \text{allowed trees}$$

Fact: The (topological) Hubbard tree is a fixed point for λ_f.

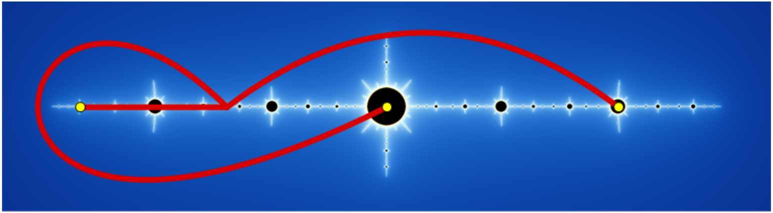
This is because the Hubbard tree H satisfies

$$H \subset f^{-1}(H).$$

Basic Algorithm: Iterate λ_f and hope to hit the Hubbard tree.

Iterated Lifting for the Airplane

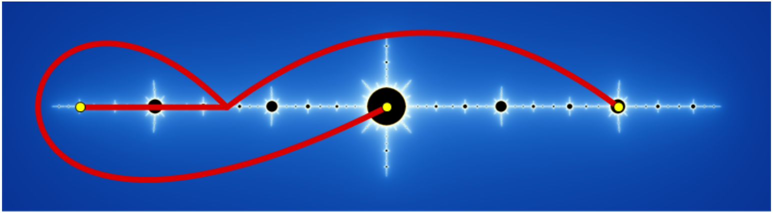
Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



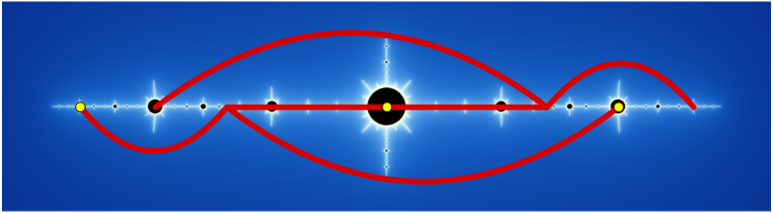
original tree T_0

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



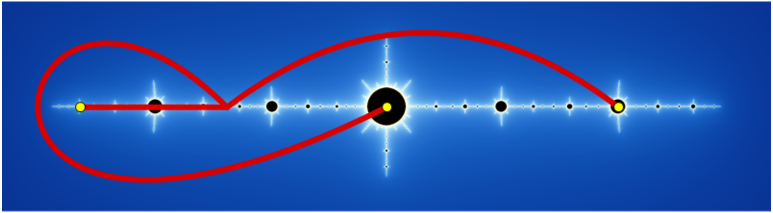
original tree T_0



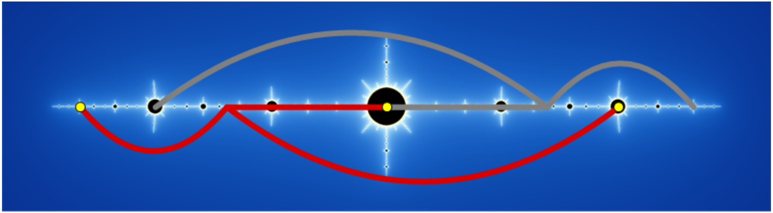
preimage $f^{-1}(T_0)$

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



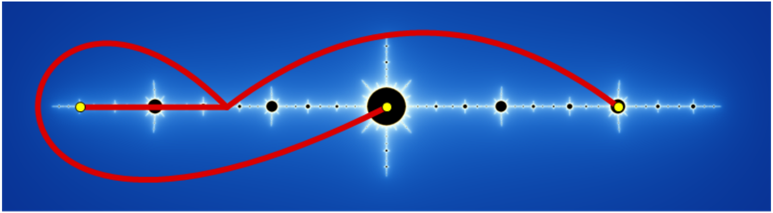
original tree T_0



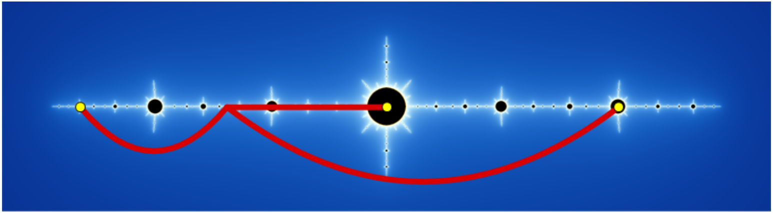
preimage $f^{-1}(T_0)$

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



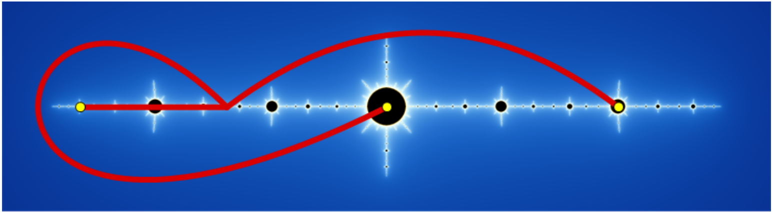
original tree T_0



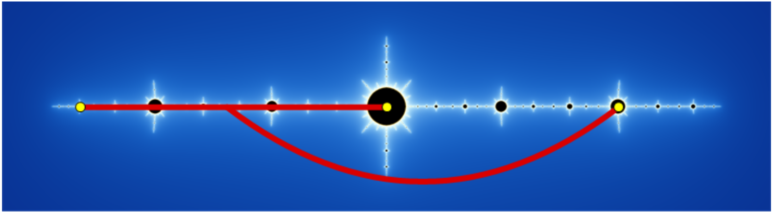
lift of T_0

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



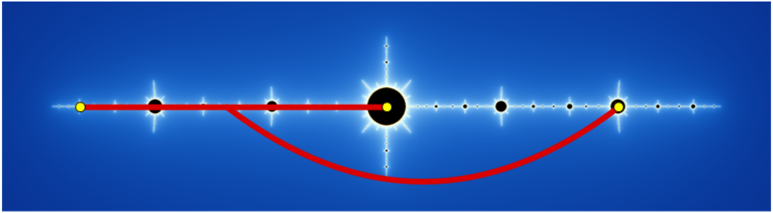
original tree T_0



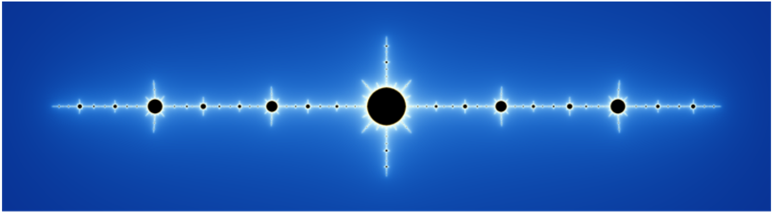
lift of T_0

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.

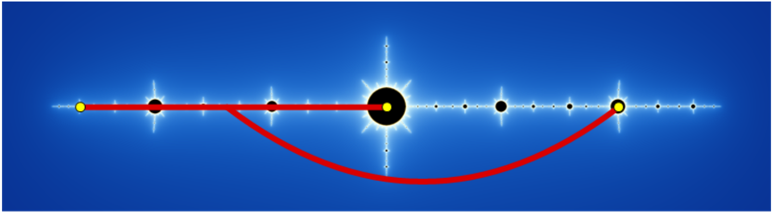


first lift T_1

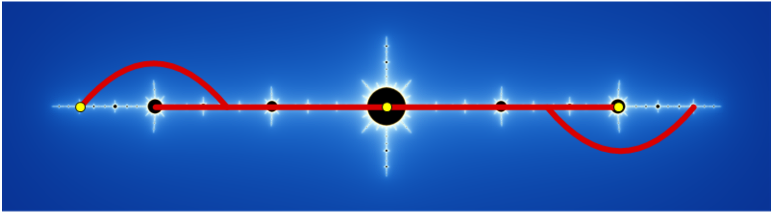


Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



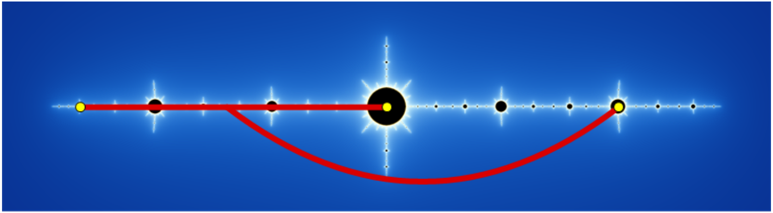
first lift T_1



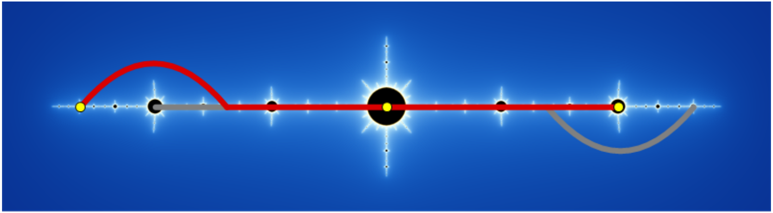
preimage $f^{-1}(T_1)$

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



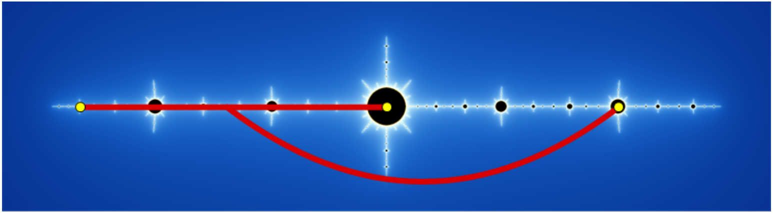
first lift T_1



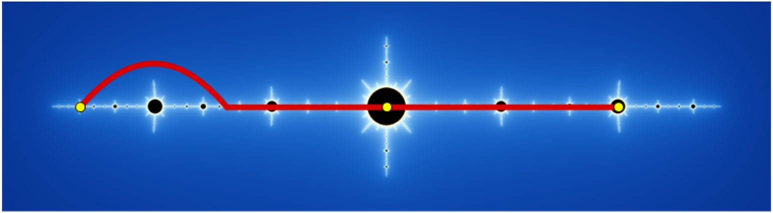
preimage $f^{-1}(T_1)$

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



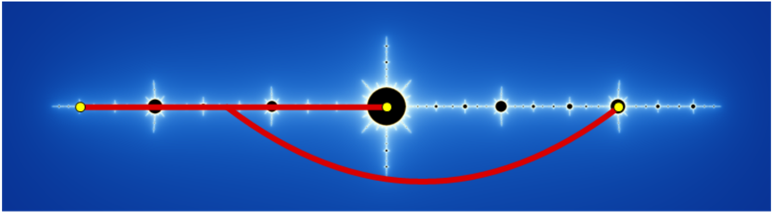
first lift T_1



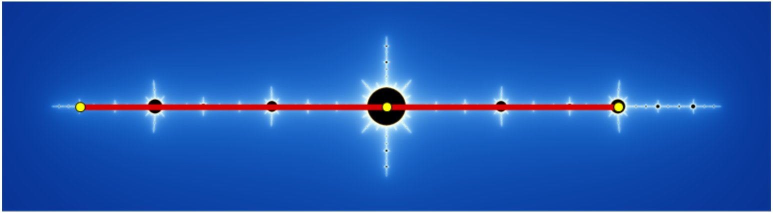
second lift T_2

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



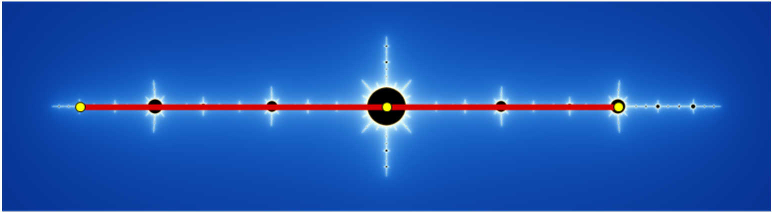
first lift T_1



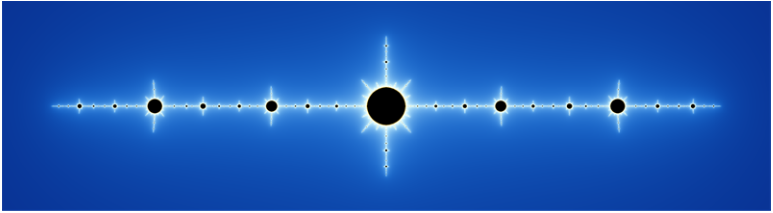
second lift T_2

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.

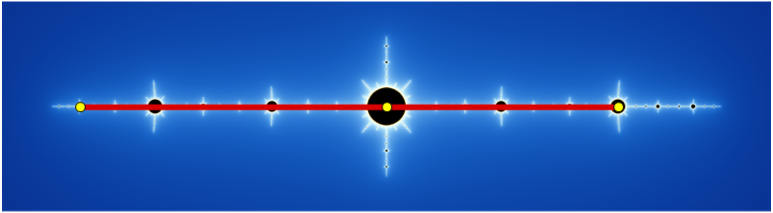


second lift T_2

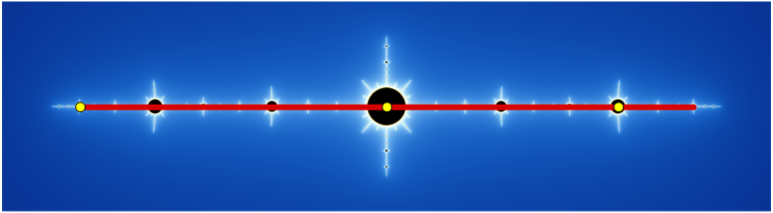


Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



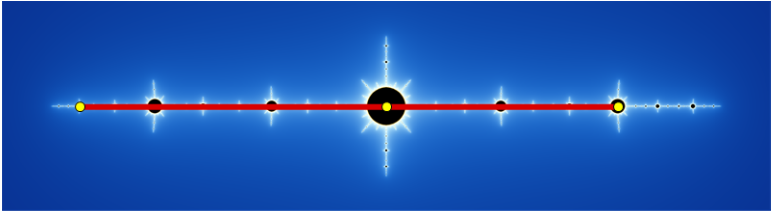
second lift T_2



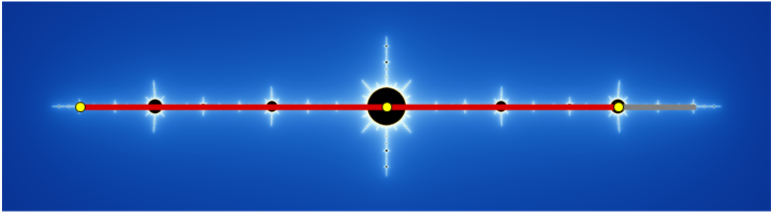
preimage $f^{-1}(T_2)$

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.



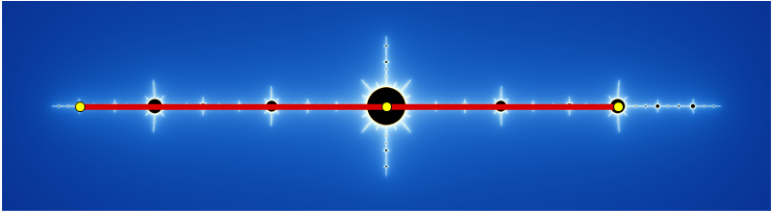
second lift T_2



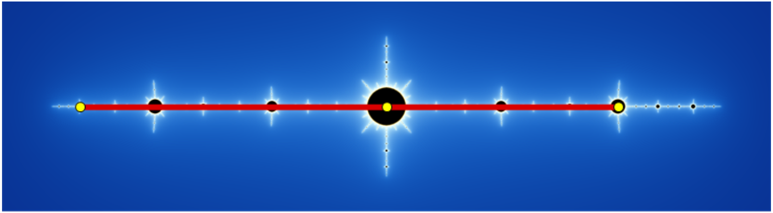
preimage $f^{-1}(T_2)$

Iterated Lifting for the Airplane

Let $f(z) \approx z^2 - 1.755$ be the airplane polynomial.

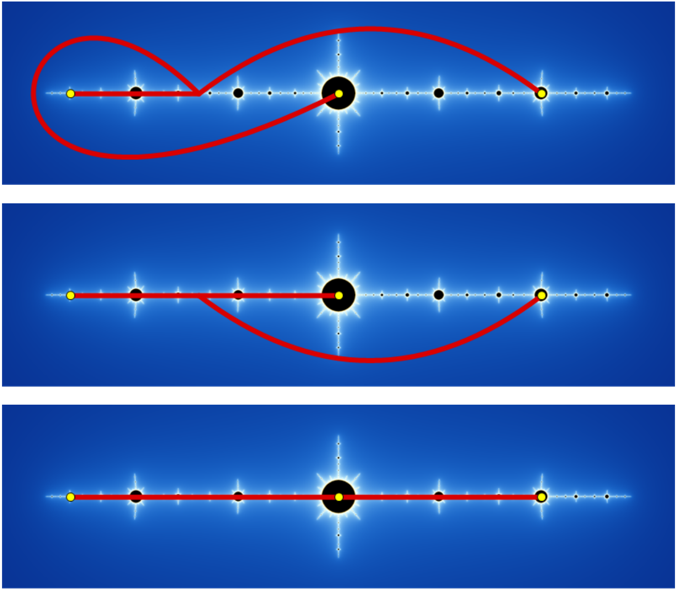


second lift T_2



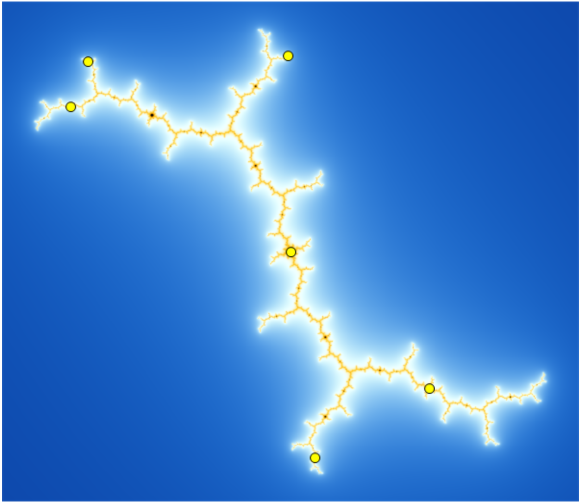
third lift T_3

Iterated Lifting for the Airplane



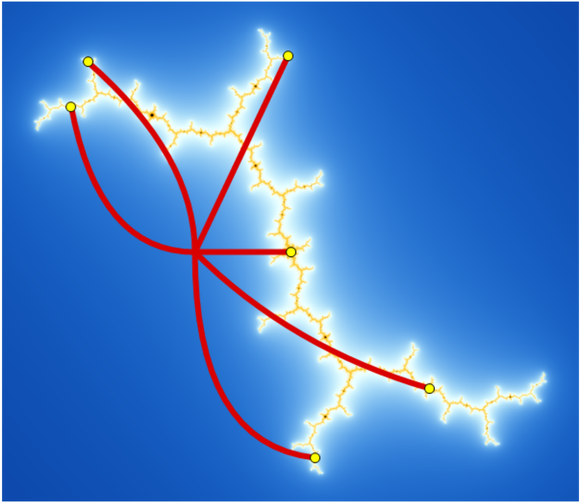
More Marked Points

Things don't get much harder with more marked points.



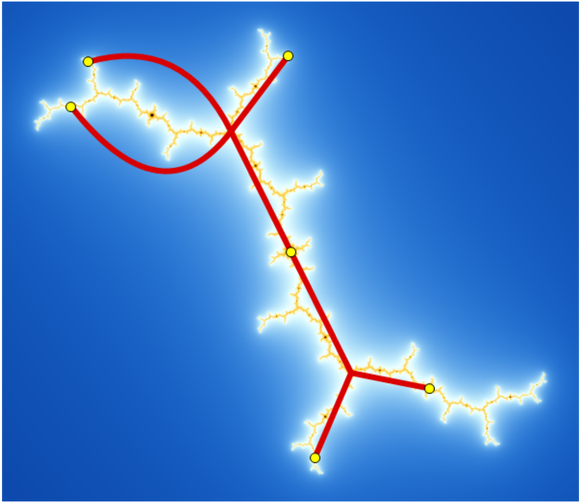
More Marked Points

Things don't get much harder with more marked points.



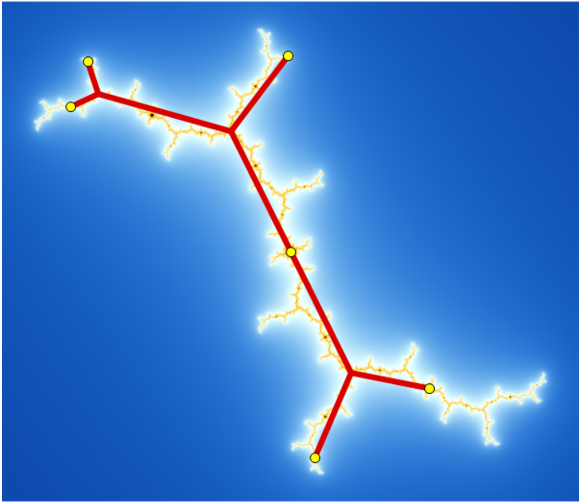
More Marked Points

Things don't get much harder with more marked points.



More Marked Points

Things don't get much harder with more marked points.

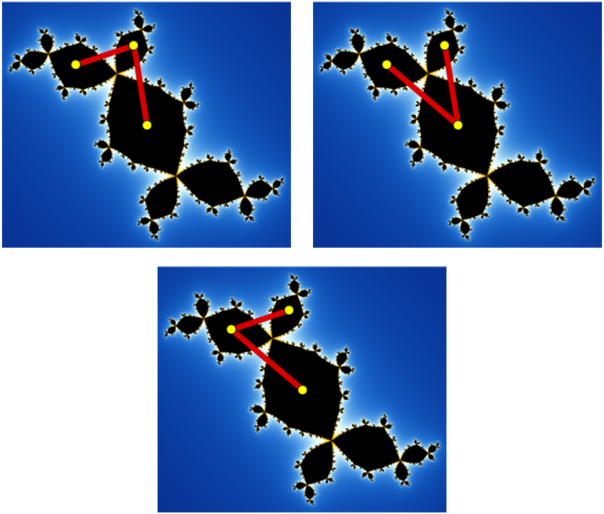


A Complication

Unfortunately, you **don't** always hit the Hubbard tree.

A Complication

Unfortunately, you **don't** always hit the Hubbard tree.



A Complication

Unfortunately, you **don't** always hit the Hubbard tree.

Theorem (BLMW 2019)

Every marked polynomial has a finite ***nucleus*** of trees that are periodic under λ_f. Iterated lifting always lands in the nucleus.

A Complication

Unfortunately, you **don't** always hit the Hubbard tree.

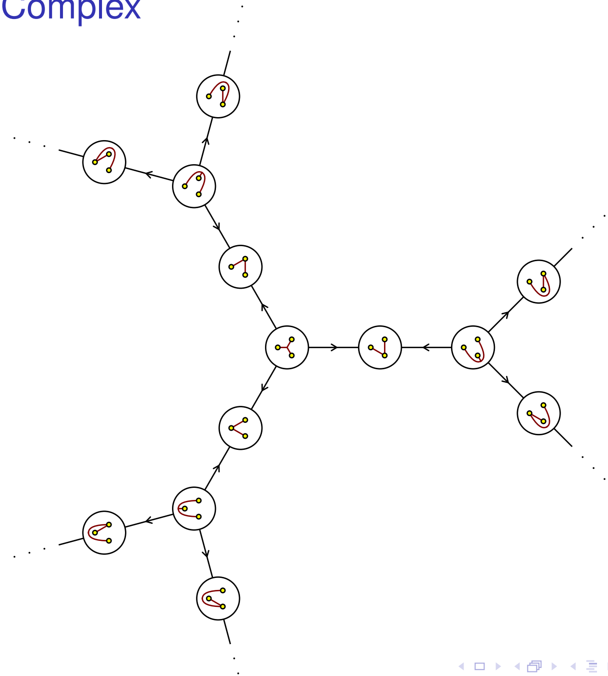
Theorem (BLMW 2019)

Every marked polynomial has a finite ***nucleus*** of trees that are periodic under λ_f. Iterated lifting always lands in the nucleus.

So the algorithm must include a resolution procedure to find the Hubbard tree once we land in the nucleus.

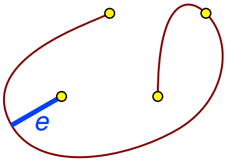
Dynamics of λ_f

The Tree Complex



Collapsing Subforests

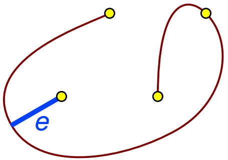
Let T be an allowed tree, and let e be an edge of T whose endpoints do not both lie in M.



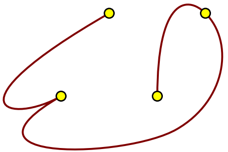
tree T

Collapsing Subforests

Let T be an allowed tree, and let e be an edge of T whose endpoints do not both lie in M.



tree T

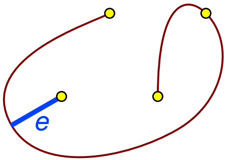


T/e

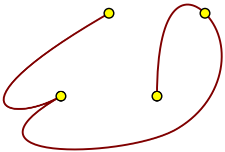
Then collapsing e to a point yields another allowed tree T/e.

Collapsing Subforests

Let T be an allowed tree, and let e be an edge of T whose endpoints do not both lie in M.



tree T



T/e

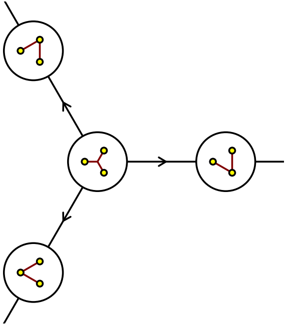
Then collapsing e to a point yields another allowed tree T/e.

More generally, we can collapse any forest in T as long as no pair of marked points are identified.

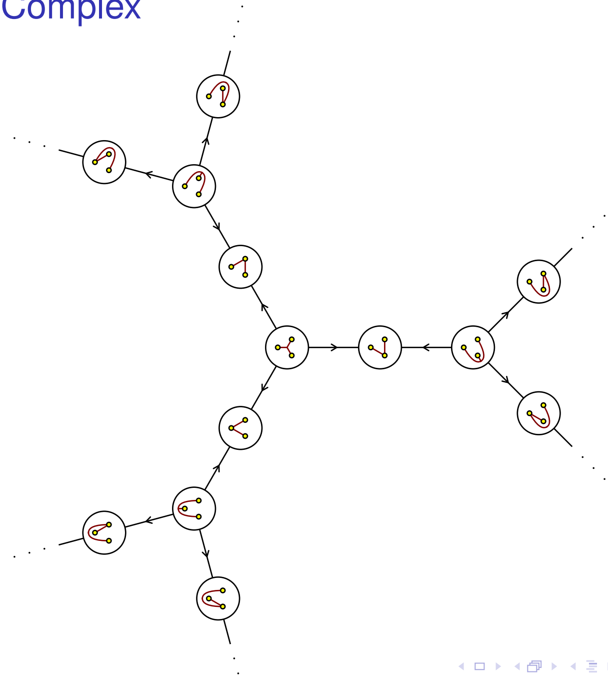
The Tree Complex

The ***tree complex*** for M has:

- One vertex for each allowed tree, and
- A directed edge $T \longrightarrow T'$ for each forest collapse.

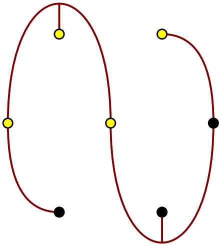


The Tree Complex

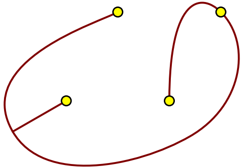


Lifting Forest Collapses

Any forest collapse $T \longrightarrow T'$ lifts to $f^{-1}(T)$.



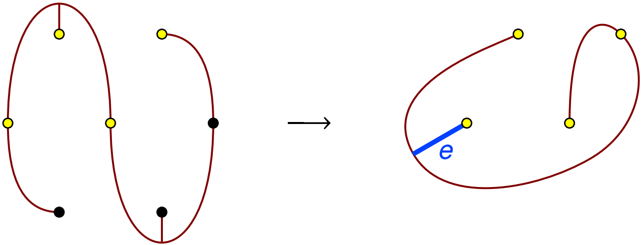
$\longrightarrow$



preimage $f^{-1}(T)$ tree T

Lifting Forest Collapses

Any forest collapse $T \longrightarrow T'$ lifts to $f^{-1}(T)$.

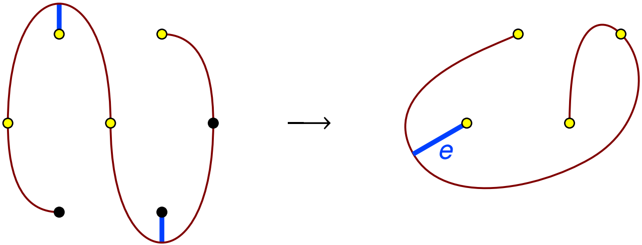


preimage $f^{-1}(T)$

tree T

Lifting Forest Collapses

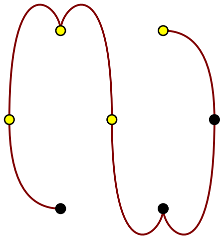
Any forest collapse $T \longrightarrow T'$ lifts to $f^{-1}(T)$.



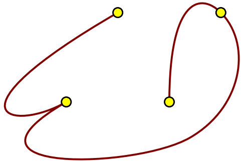
preimage $f^{-1}(T)$ tree T

Lifting Forest Collapses

Any forest collapse $T \longrightarrow T'$ lifts to $f^{-1}(T)$.



$\longrightarrow$

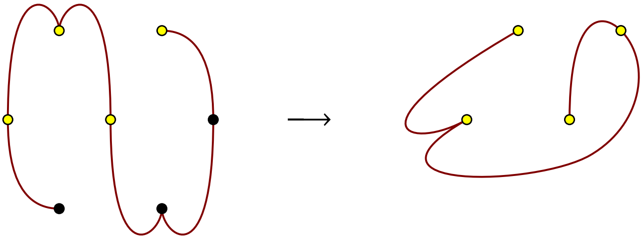


preimage $f^{-1}(T')$

collapsed tree T'

Lifting Forest Collapses

Any forest collapse $T \longrightarrow T'$ lifts to $f^{-1}(T)$.



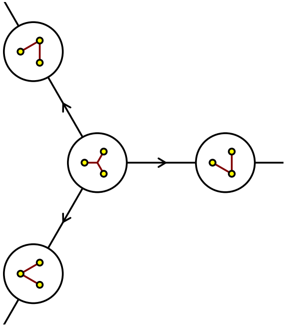
preimage $f^{-1}(T')$ collapsed tree T'

It follows that either

$$\lambda_f(T) \to \lambda_f(T') \qquad \text{or} \qquad \lambda_f(T) = \lambda_f(T').$$

The Tree Complex

So λ_f induces a non-expanding map on the tree complex. This is the ***lifting map***.



The Tree Complex

So λ_f induces a non-expanding map on the tree complex. This is the ***lifting map***.

Theorem (BLMW 2019)

If f is a polynomial, then every allowed tree is either periodic or pre-periodic under λ_f.

The Tree Complex

So λ_f induces a non-expanding map on the tree complex. This is the ***lifting map***.

Theorem (BLMW 2019)

If f is a polynomial, then every allowed tree is either periodic or pre-periodic under λ_f.

Proof.

Since the Hubbard tree H is fixed and λ_f is non-expanding, each ball in the complex centered at H maps into itself. Such a ball has finitely many trees. $\square$

The Tree Complex

So λ_f induces a non-expanding map on the tree complex. This is the ***lifting map***.

Theorem (BLMW 2019)

If f is a polynomial, then every allowed tree is either periodic or pre-periodic under λ_f.

The Tree Complex

So λ_f induces a non-expanding map on the tree complex. This is the ***lifting map***.

Theorem (BLMW 2019)

If f is a polynomial, then every allowed tree is either periodic or pre-periodic under λ_f.

The set of all periodic trees is the ***nucleus*** for λ_f.

The Tree Complex

So λ_f induces a non-expanding map on the tree complex. This is the ***lifting map***.

Theorem (BLMW 2019)

If f is a polynomial, then every allowed tree is either periodic or pre-periodic under λ_f.

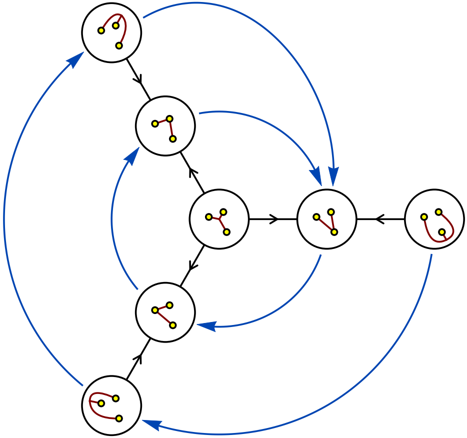
The set of all periodic trees is the ***nucleus*** for λ_f.

Theorem (BLMW 2019)

The nucleus for λ_f lies within the 2-ball centered at H.

Example: The Rabbit Nucleus

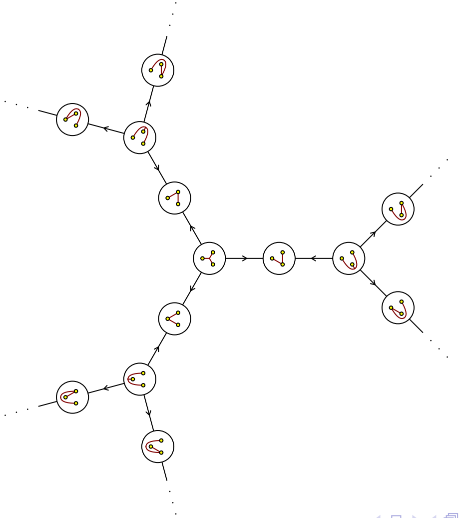
The nucleus for the rabbit is the 1-neighborhood of the Hubbard tree.



What's Going On?

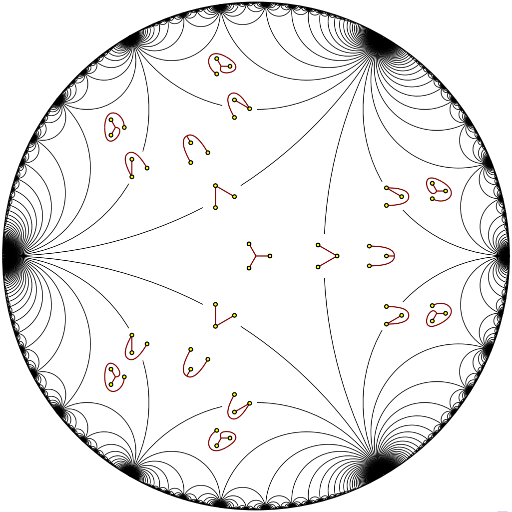
What's Going On?

The tree complex is actually the spine of a certain simplicial subdivision of Teichmüller space (discovered by Penner).



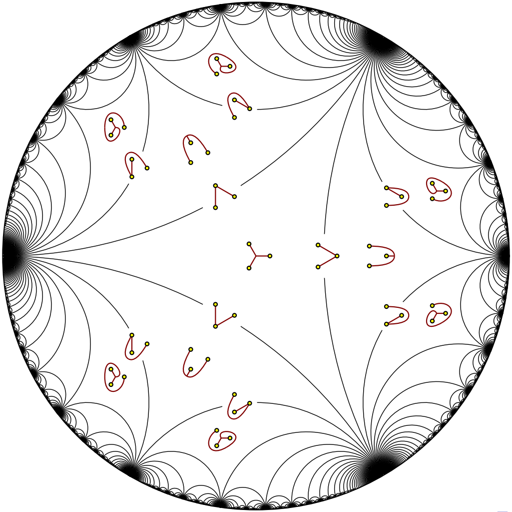
What's Going On?

The tree complex is actually the spine of a certain simplicial subdivision of Teichmüller space (discovered by Penner).



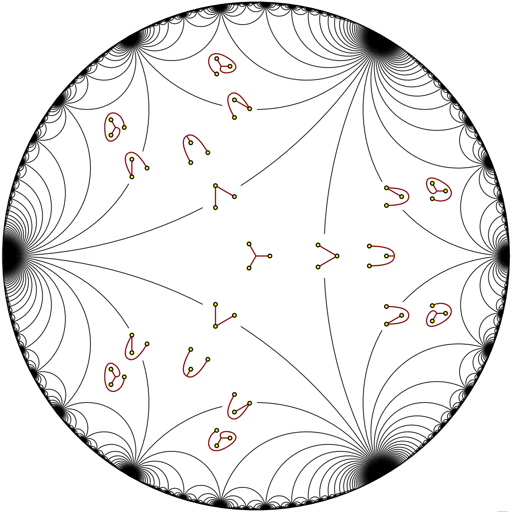
What's Going On?

Each tree corresponds to an open simplex. Different points in the simplex correspond to different metrics on the tree.



What's Going On?

The lifting map λ_f seems to be a combinatorial version of Thurston's pullback map.



Finding the Hubbard Tree

The Story So Far

So far: We can iterate lifting until we find a periodic tree.

This gets us within 2 steps of the Hubbard tree.

Questions

1. How do we get to the Hubbard tree itself?
2. How would we even recognize the Hubbard tree if we found it?

Invariant Trees

An allowed tree T is ***invariant*** if $\lambda_f(T) = T$. Up to isotopy, such a tree satisfies

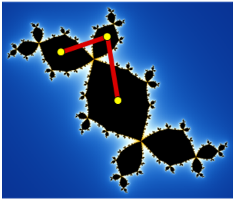
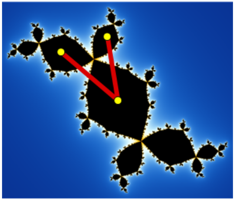
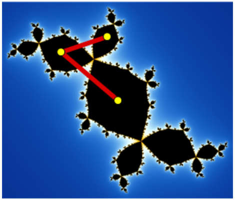
$$T \subset f^{-1}(T).$$

Invariant Trees

An allowed tree T is ***invariant*** if $\lambda_f(T) = T$. Up to isotopy, such a tree satisfies

$$T \subset f^{-1}(T).$$

Note that periodic trees are invariant for f^k.

Invariant Trees

An allowed tree T is ***invariant*** if $\lambda_f(T) = T$. Up to isotopy, such a tree satisfies

$$T \subset f^{-1}(T).$$

Note that periodic trees are invariant for f^k.

Question

How do we tell whether an invariant tree T is the Hubbard tree?

Invariant Trees

An allowed tree T is ***invariant*** if $\lambda_f(T) = T$. Up to isotopy, such a tree satisfies

$$T \subset f^{-1}(T).$$

Note that periodic trees are invariant for f^k.

Question

How do we tell whether an invariant tree T is the Hubbard tree?

Answer

It suffices for there to exist *any* polynomial with Hubbard tree T that induces the same mapping $f^{-1}(T) \to T$.

Poirier's Conditions

Alfredo Poirier completely classified possible Hubbard trees in 1993.

Theorem (Poirier's Conditions)

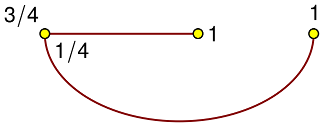
An invariant tree T for (f, M) is a topological Hubbard tree if and only if the following conditions are satisfied:

1. **(Angle Condition)** T has an invariant angle assignment.
2. **(Expanding Condition)** Every forward-invariant subforest of T contains a critical point.

The Angle Condition

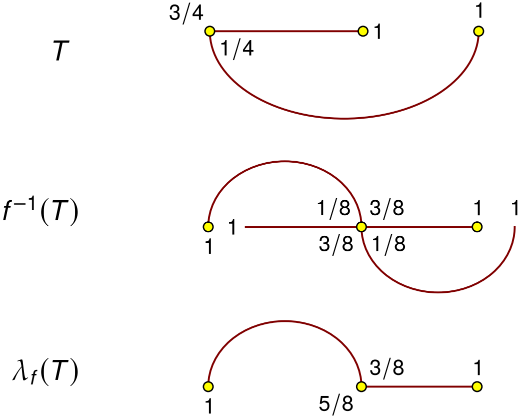
Here is an ***angle assignment*** for a tree T.

T



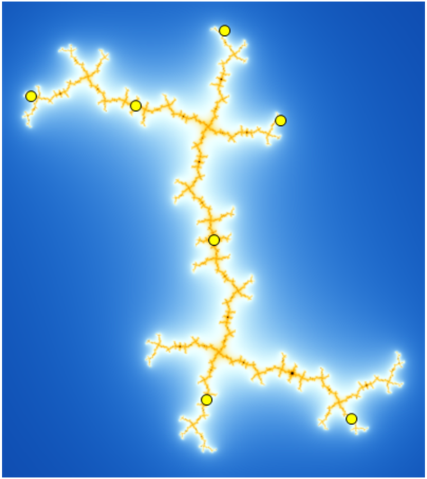
The Angle Condition

We can ***lift*** the angle assignment to $\lambda_f(T)$.



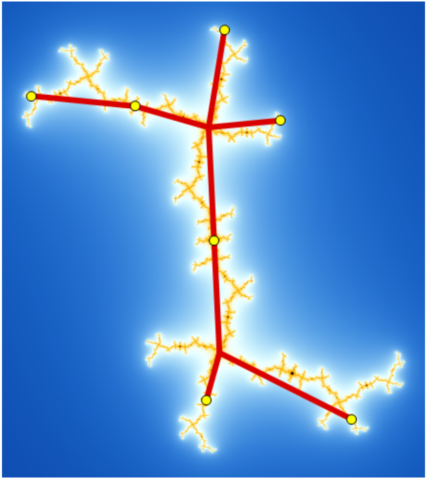
The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.



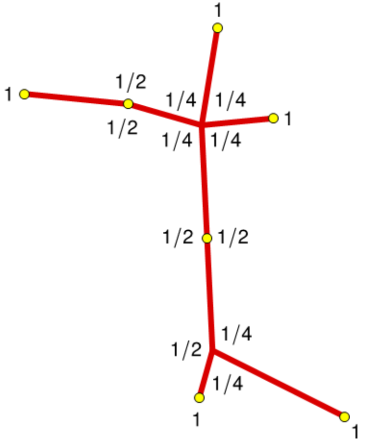
The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.



The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.



The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.

Theorem (BLMW 2019)

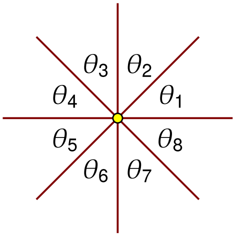
Every invariant tree is adjacent to an invariant tree that satisfies the angle condition.

The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.

Theorem (BLMW 2019)

Every invariant tree is adjacent to an invariant tree that satisfies the angle condition.

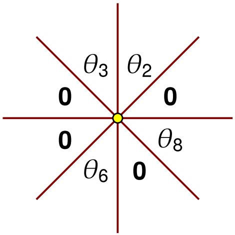


The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.

Theorem (BLMW 2019)

Every invariant tree is adjacent to an invariant tree that satisfies the angle condition.

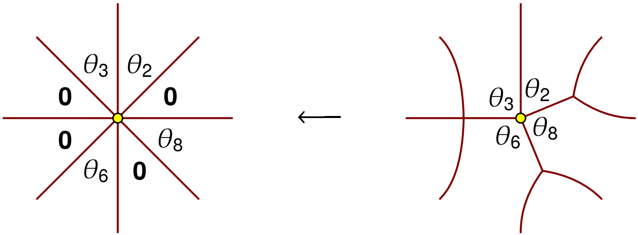


The Angle Condition

An invariant tree satisfies the ***angle condition*** if there exists an angle assignment that lifts to itself.

Theorem (BLMW 2019)

Every invariant tree is adjacent to an invariant tree that satisfies the angle condition.



The Expanding Condition

Let T be an invariant tree for (f, M).

A subforest $S \subset T$ is ***forward invariant*** if $f(S) \subset S$. The tree T satisfies the ***expanding condition*** if every such S contains a critical point.

Theorem (BLMW 2019)

Every invariant tree that satisfies the angle condition is adjacent to the Hubbard tree.

Proof.

Collapse the unique maximal invariant forest that contains no critical points. $\square$

The Algorithm

So given an (f, M), the algorithm is as follows:

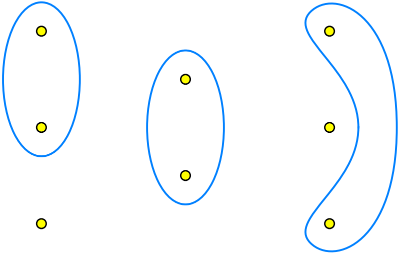
1. Start with any allowed tree and iterate lifting until you find a periodic tree T.
2. Check if T satisfies the angle condition. If it doesn't, move to an adjacent tree T' that does.
3. Check if T' satisfies the expanding condition. If it doesn't, move to an adjacent tree T'' that does.

Then T'' is the topological Hubbard tree.

The Obstructed Case

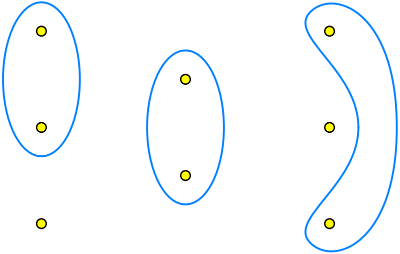
The Canonical Obstruction

Every obstructed (f, M) has a special collection of curves called the ***canonical obstruction***.



The Canonical Obstruction

Every obstructed (f, M) has a special collection of curves called the ***canonical obstruction***.

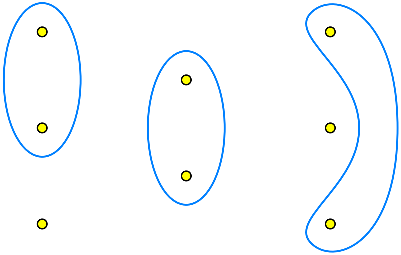


These are the curves whose hyperbolic lengths go to zero.

Pilgrim (2001) proved that the canonical obstruction is fully invariant under f, and is a Thurston obstruction.

The Canonical Obstruction

Every obstructed (f, M) has a special collection of curves called the ***canonical obstruction***.

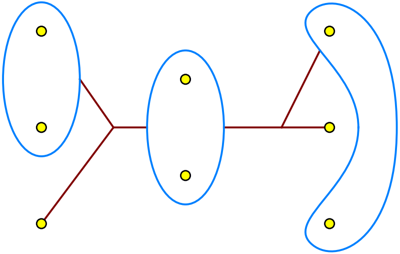


The curves of the canonical obstruction bound disjoint disks.

Selinger (2013) proved that the map on the exterior is topologically equivalent to a polynomial.

The Canonical Obstruction

Every obstructed (f, M) has a special collection of curves called the ***canonical obstruction***.

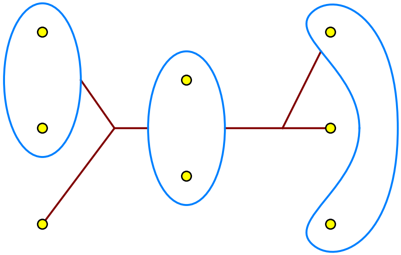


The curves of the canonical obstruction bound disjoint disks.

Selinger (2013) proved that the map on the exterior is topologically equivalent to a polynomial.

The Canonical Obstruction

Every obstructed (f, M) has a special collection of curves called the ***canonical obstruction***.



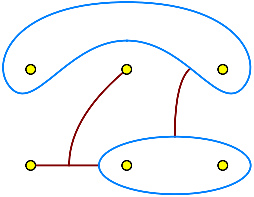
We call this the ***Hubbard bubble tree*** for the obstructed map.

When (f, M) is obstructed, we can use the tree lifting algorithm to find the Hubbard bubble tree.

Finding the Hubbard Bubble Tree

In general, a ***bubble tree*** consists of:

1. Finitely many essential curves in $(\mathbb{C}, M)$ with disjoint interiors.
2. A tree on the exterior of these curves.

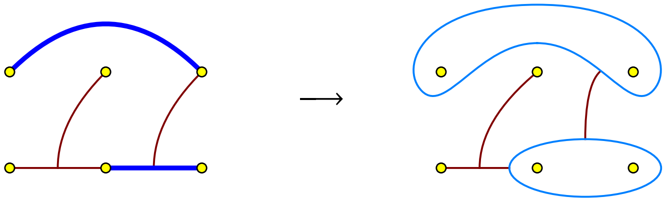


Finding the Hubbard Bubble Tree

In general, a ***bubble tree*** consists of:

1. Finitely many essential curves in $(\mathbb{C}, M)$ with disjoint interiors.
2. A tree on the exterior of these curves.

Bubble trees can be obtained from trees by collapsing subforests.



Finding the Hubbard Bubble Tree

In general, a ***bubble tree*** consists of:

1. Finitely many essential curves in $(\mathbb{C}, M)$ with disjoint interiors.
2. A tree on the exterior of these curves.

Bubble trees can be obtained from trees by collapsing subforests.

This defines the ***augmented tree complex***.

Finding the Hubbard Bubble Tree

In general, a ***bubble tree*** consists of:

1. Finitely many essential curves in $(\mathbb{C}, M)$ with disjoint interiors.
2. A tree on the exterior of these curves.

Bubble trees can be obtained from trees by collapsing subforests. This defines the ***augmented tree complex***.

Theorem (BLMW 2019)

For an obstructed (f, M), the sequence of lifts eventually lands in the 2-neighborhood of the Hubbard bubble tree in the augmented complex.

The End