Fighting Fish:

enumerative properties

ENRICA DUCHI
IRIF, Université Paris Diderot

VERONICA GUERRINI
& SIMONE RINALDI
DIISM, Università di Siena

GILLES SCHAEFFER
LIX, CNRS and École Polytechnique

Séminaire Philippe Flajolet, 2017

Summary of the talk

Fighting fish, a new combinatorial model
of discrete branching surfaces

Exact counting formulas for fighting fish

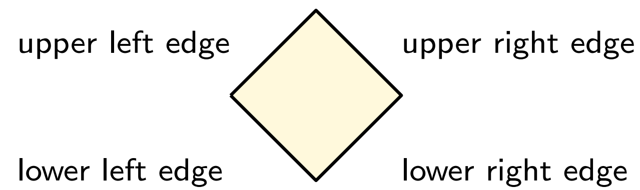
Decompositions for fighting fish

Fighting fish VS classical combinatorial structures
a bijective challenge...

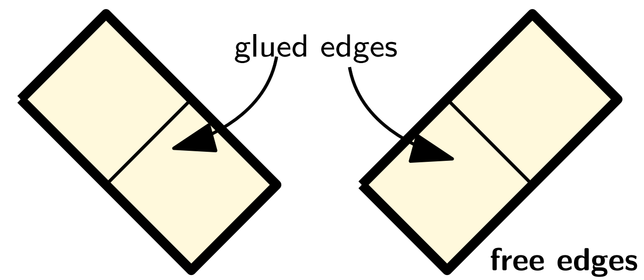
Fighting fish, definition

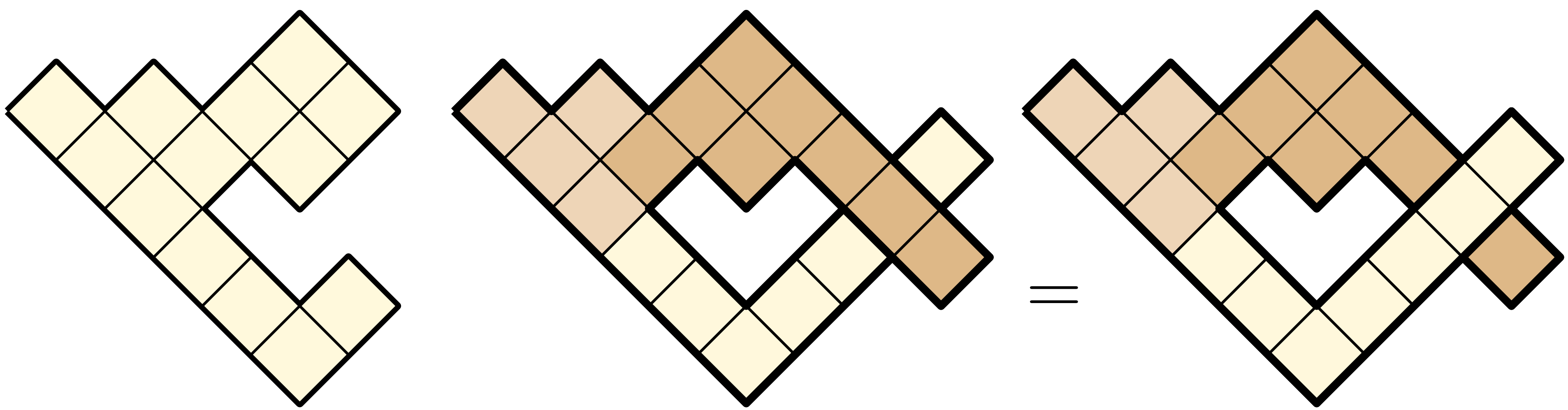
Cells

45^o tilted unit square
(of thin paper or cloth)



Build surface by gluing cells along edges in a coherent way: upper left with lower right or lower left with upper right.

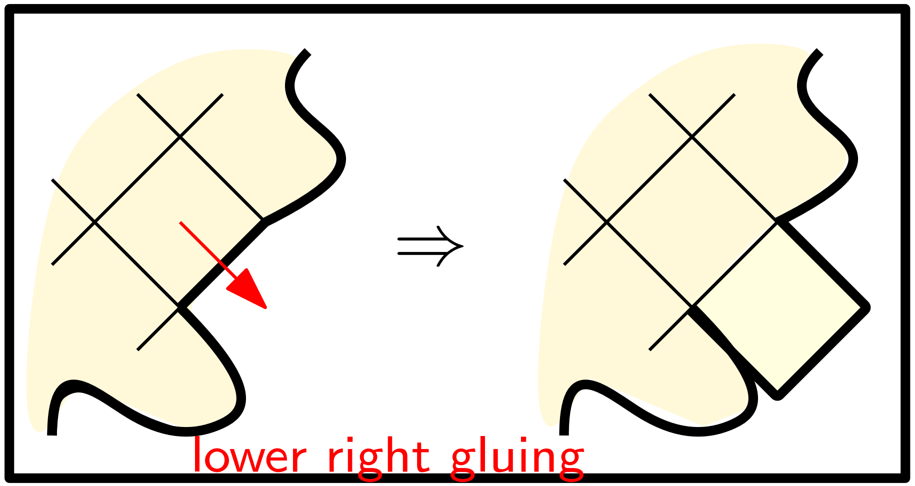


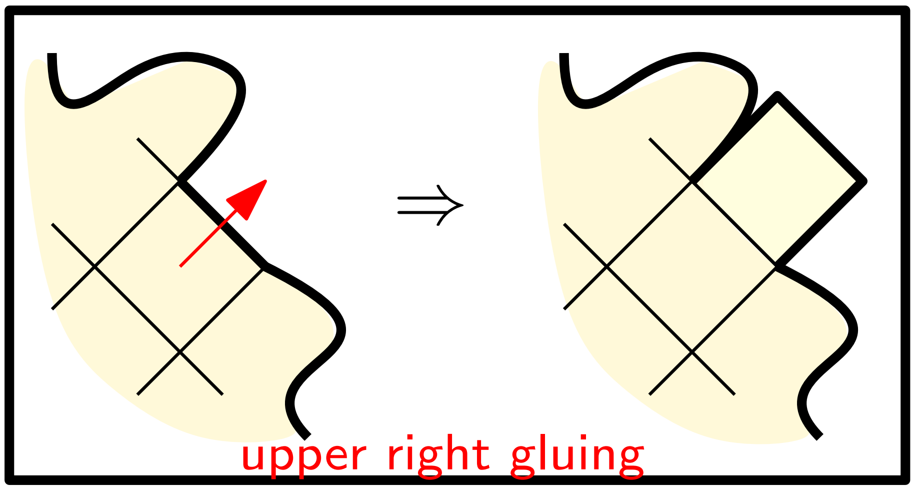


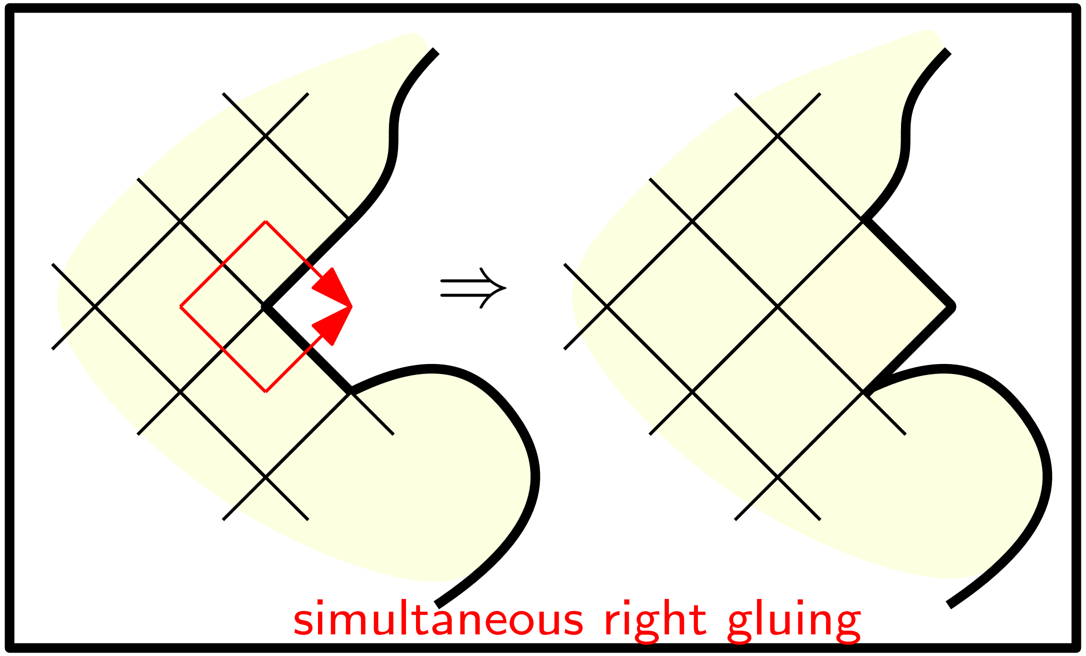
These objects do not necessarily fit in the plane so my pictures are projections of the actual surfaces: Apparently overlapping cells are in fact independant.

Fighting fish, definition

Directed cell aggregation. Restrict to only three legal ways to add cells: by lower right gluing, upper right gluing, or simultaneous lower and upper right gluings from adjacent free edges.



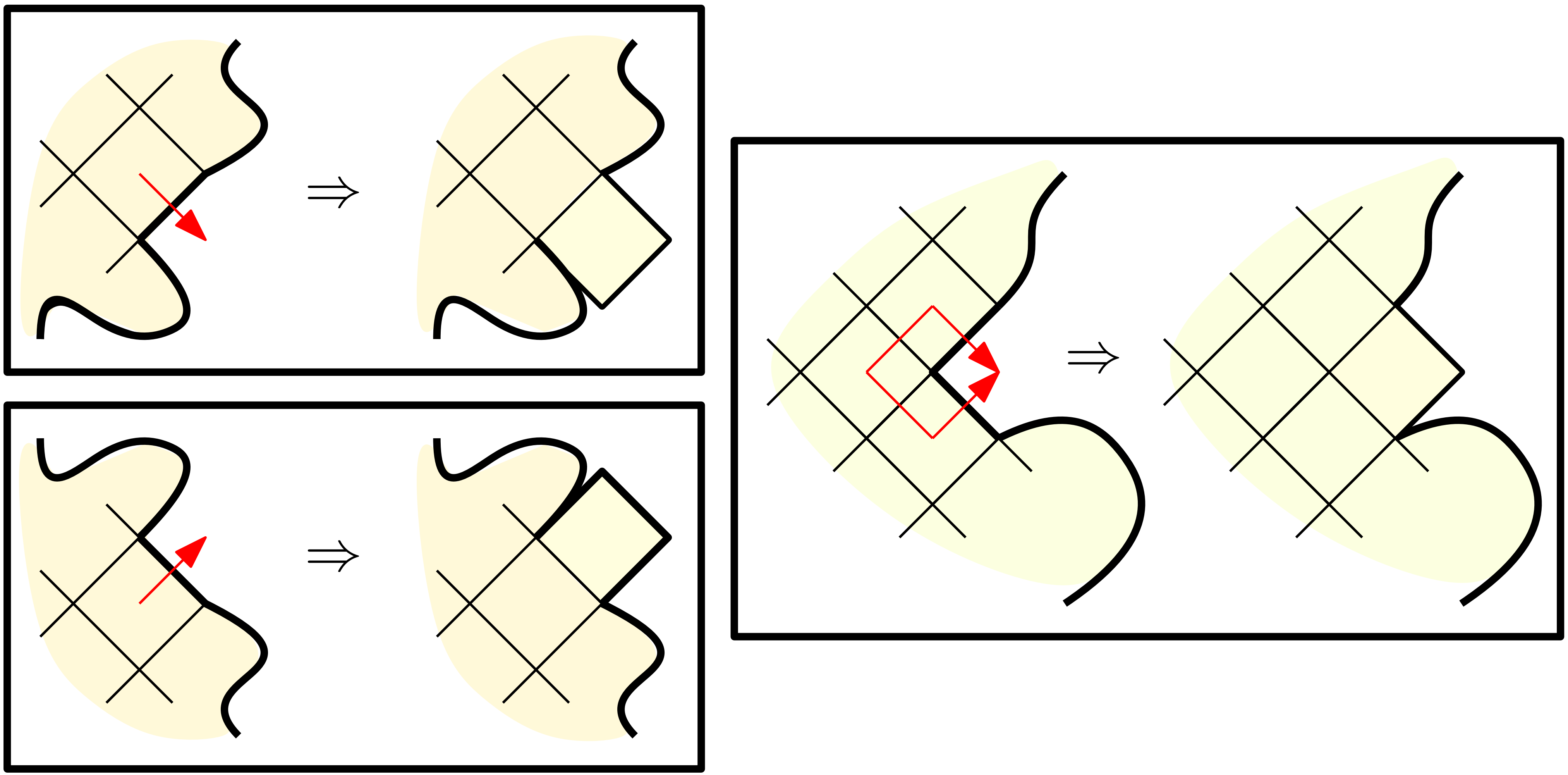




Fighting fish, definition

Lemma. Single cell + aggregations
$\Rightarrow$ a simply connected surface

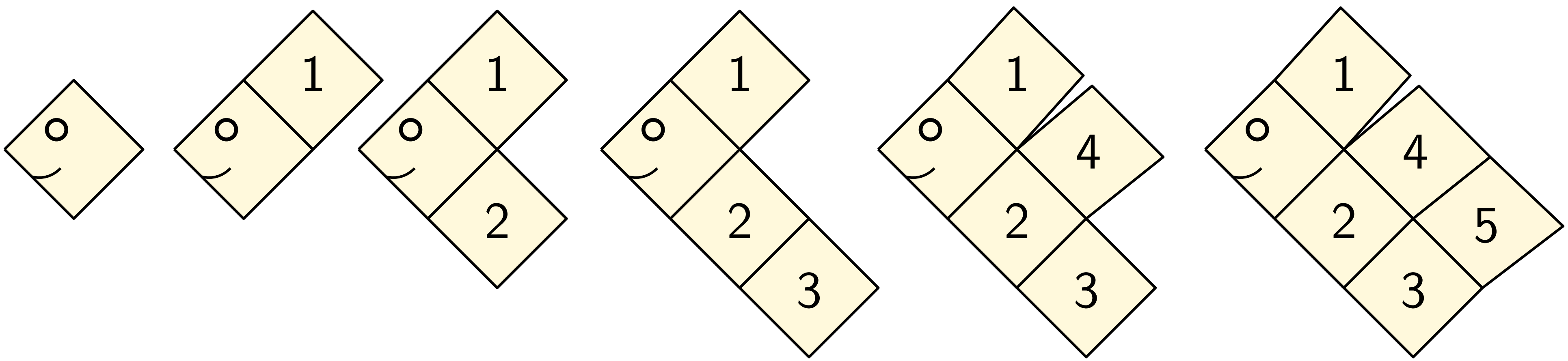
Remark. Such surfaces can be recovered from their boundary walk.



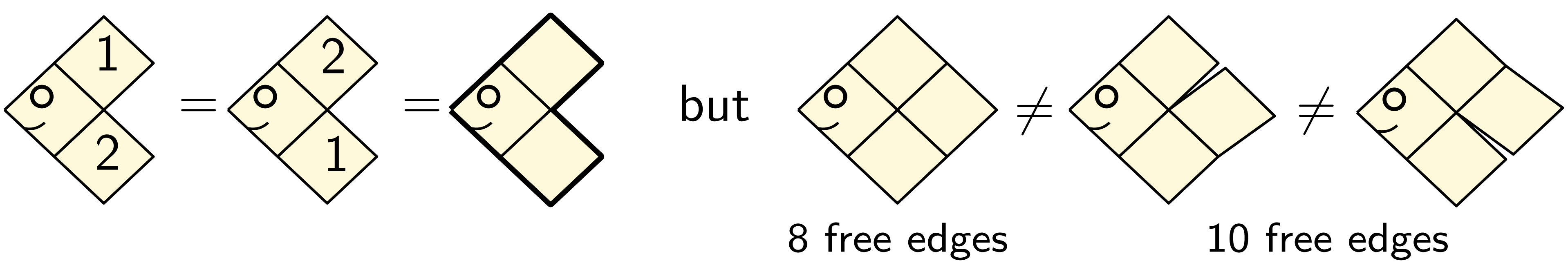
Fighting fish, definition

Fighting fish

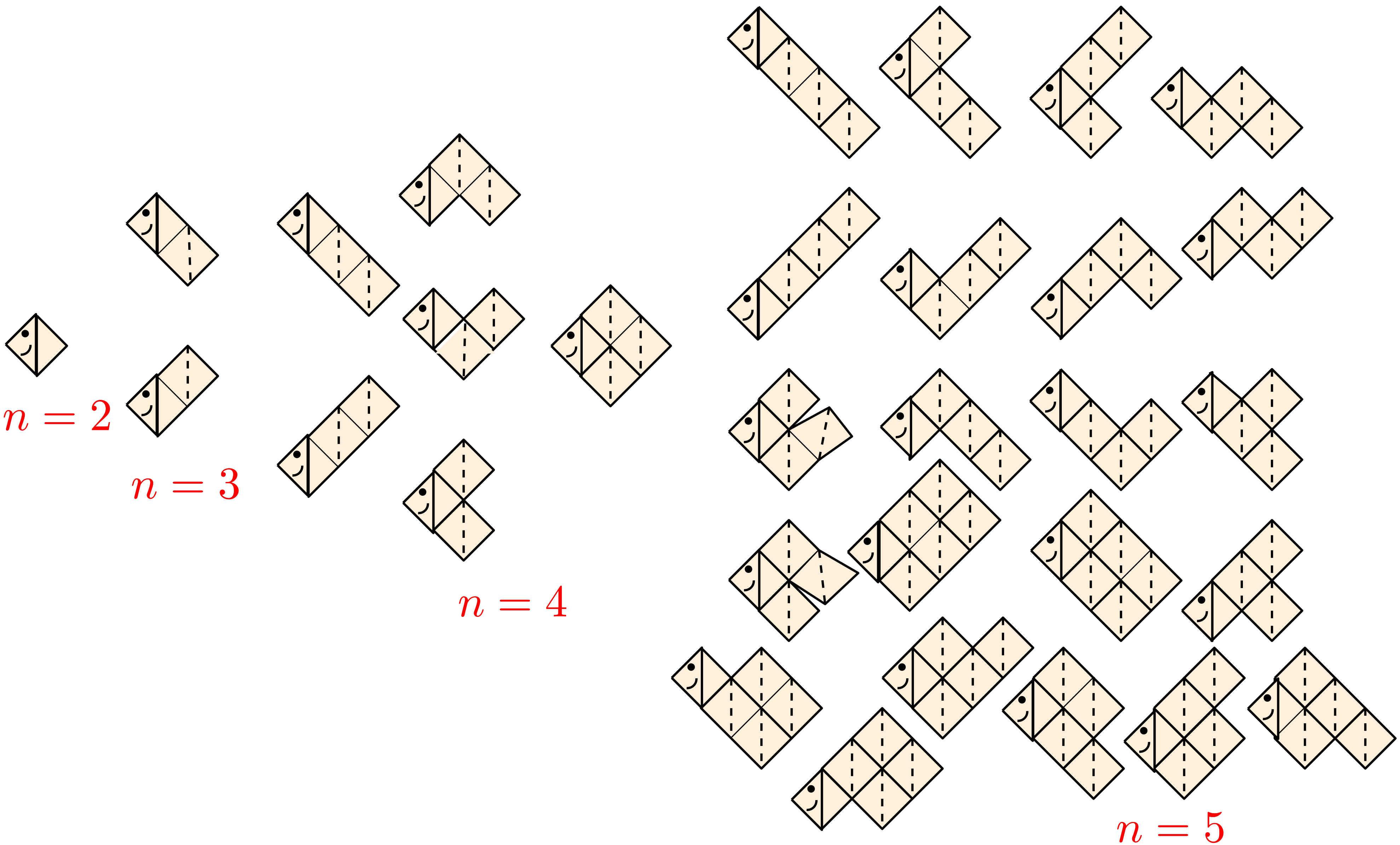
A fighting fish is a surface that can be obtained from a single cell by a sequence of directed cell agregations.



We are interested only in the resulting surface, not in the aggregation order (but type of aggregation matters)

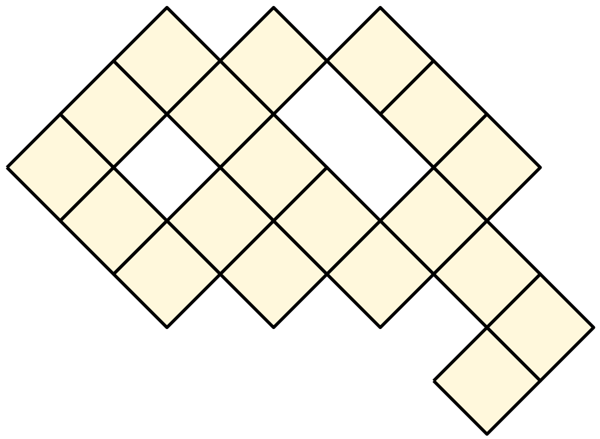


Small fighting fish



Fighting fish versus polyominoes

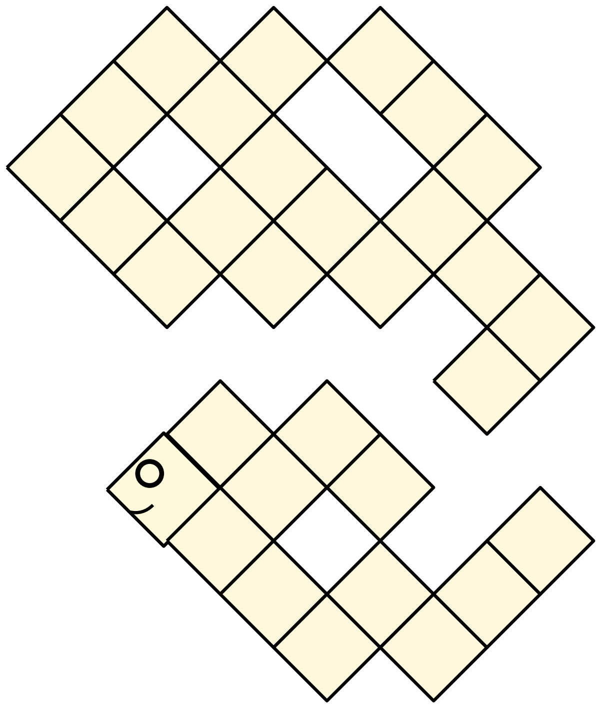
Polyomino = edge-connected set of cells **of the planar square lattice**



Fighting fish versus polyominoes

Polyomino = edge-connected set of cells **of the planar square lattice**

Directed polyominoes: there is a cell, the head, from which all cells can be reached by a left-to-right path.



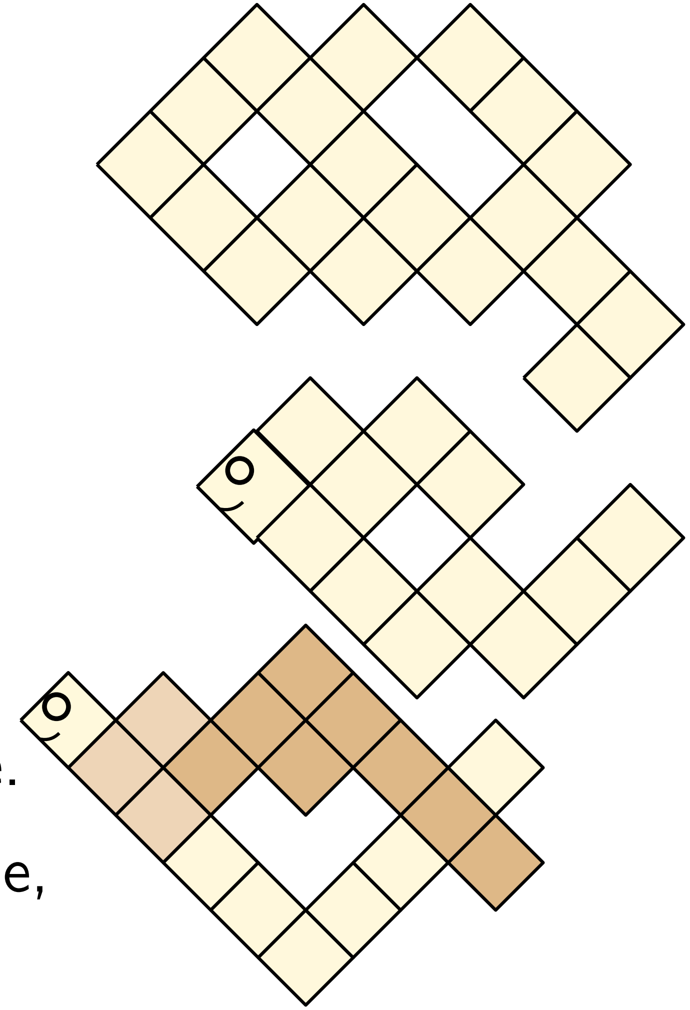
Fighting fish versus polyominoes

Polyomino $=$ edge-connected set of cells **of the planar square lattice**

Directed polyominoes: there is a cell, the head, from which all cells can be reached by a left-to-right path.

Proposition.
A fighting fish is a directed polyomino **iff** its projection in the plane is injective.

$\Rightarrow$ fighting fish do not all fit in the plane, *ie* they are not all polyominoes.



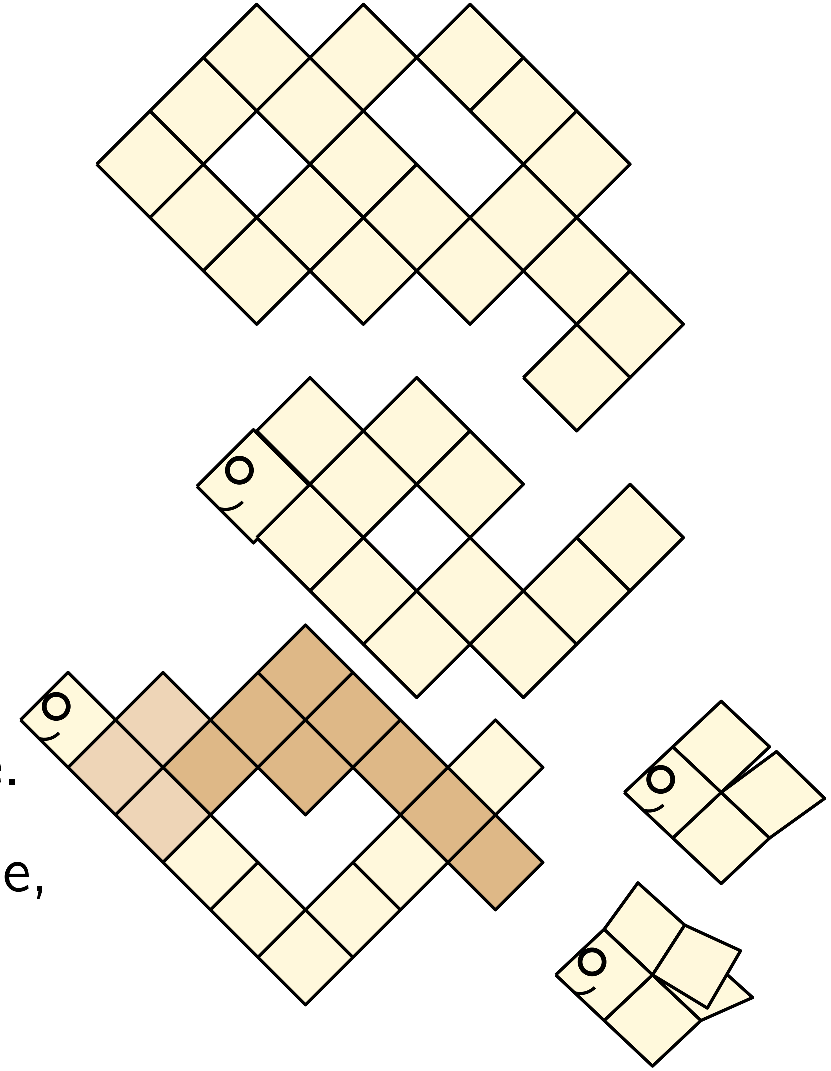
Fighting fish versus polyominoes

Polyomino = edge-connected set of cells **of the planar square lattice**

Directed polyominoes: there is a cell, the head, from which all cells can be reached by a left-to-right path.

Proposition.
A fighting fish is a directed polyomino **iff** its projection in the plane is injective.

$\Rightarrow$ fighting fish do not all fit in the plane, *ie* they are not all polyominoes.



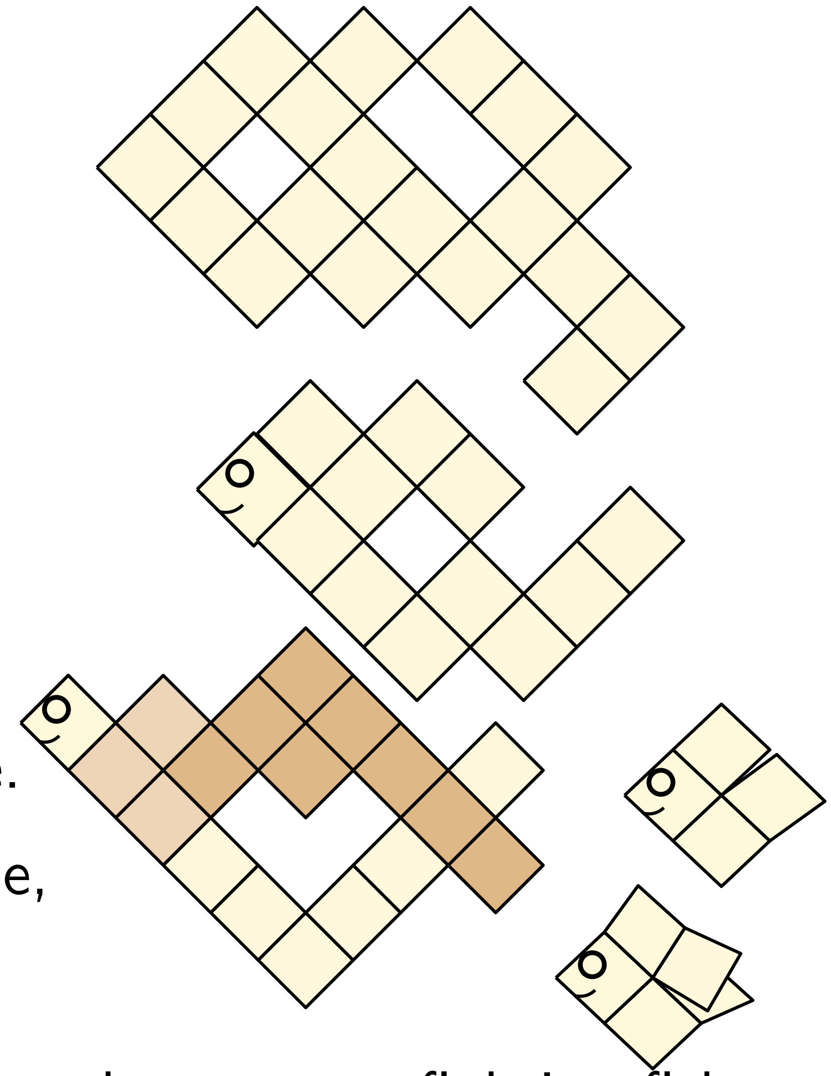
Fighting fish versus polyominoes

Polyomino = edge-connected set of cells **of the planar square lattice**

Directed polyominoes: there is a cell, the head, from which all cells can be reached by a left-to-right path.

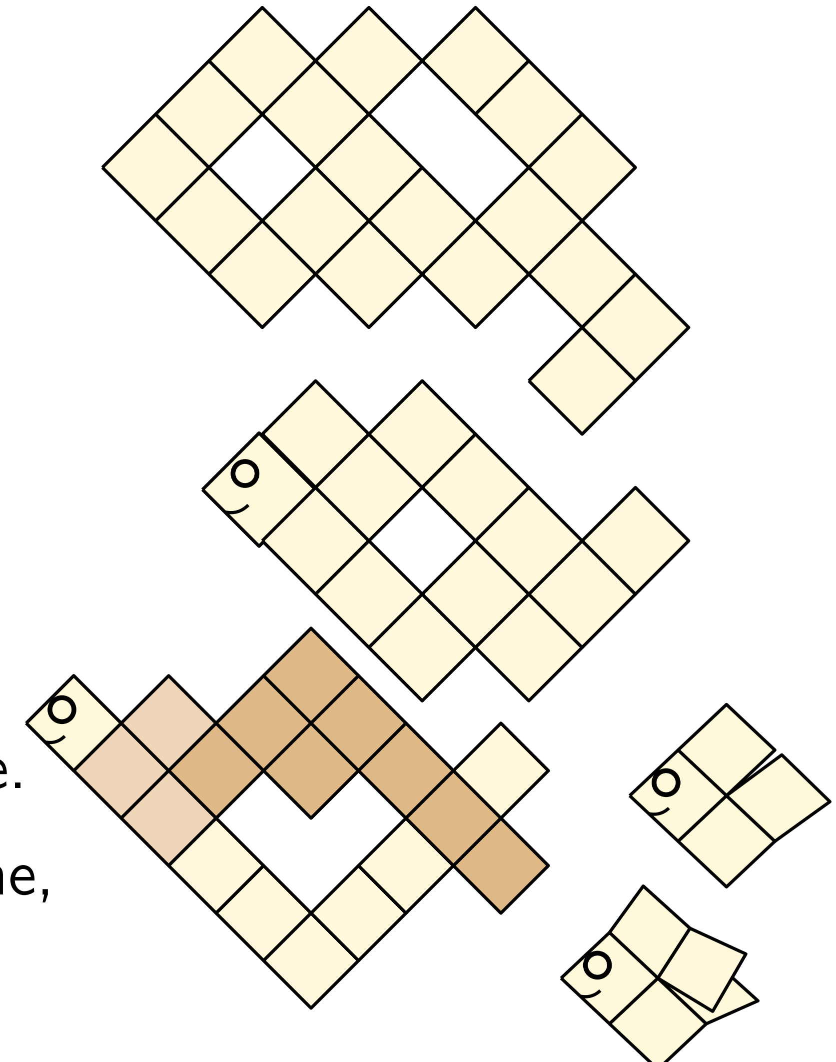
Proposition.
A fighting fish is a directed polyomino **iff** its projection in the plane is injective.

$\Rightarrow$ fighting fish do not all fit in the plane, *ie* they are not all polyominoes.



Conversely there are directed polyominoes that are not fighting fish:

Fighting fish versus polyominoes



Polyomino = edge-connected set of cells **of the planar square lattice**

Directed polyominoes: there is a cell, the head, from which all cells can be reached by a left-to-right path.

Proposition.
A fighting fish is a directed polyomino **iff** its projection in the plane is injective.

$\Rightarrow$ fighting fish do not all fit in the plane, *ie* they are not all polyominoes.

Conversely there are directed polyominoes that are not fighting fish:
Fighting fish are a generalization of directed polyominoes without holes.

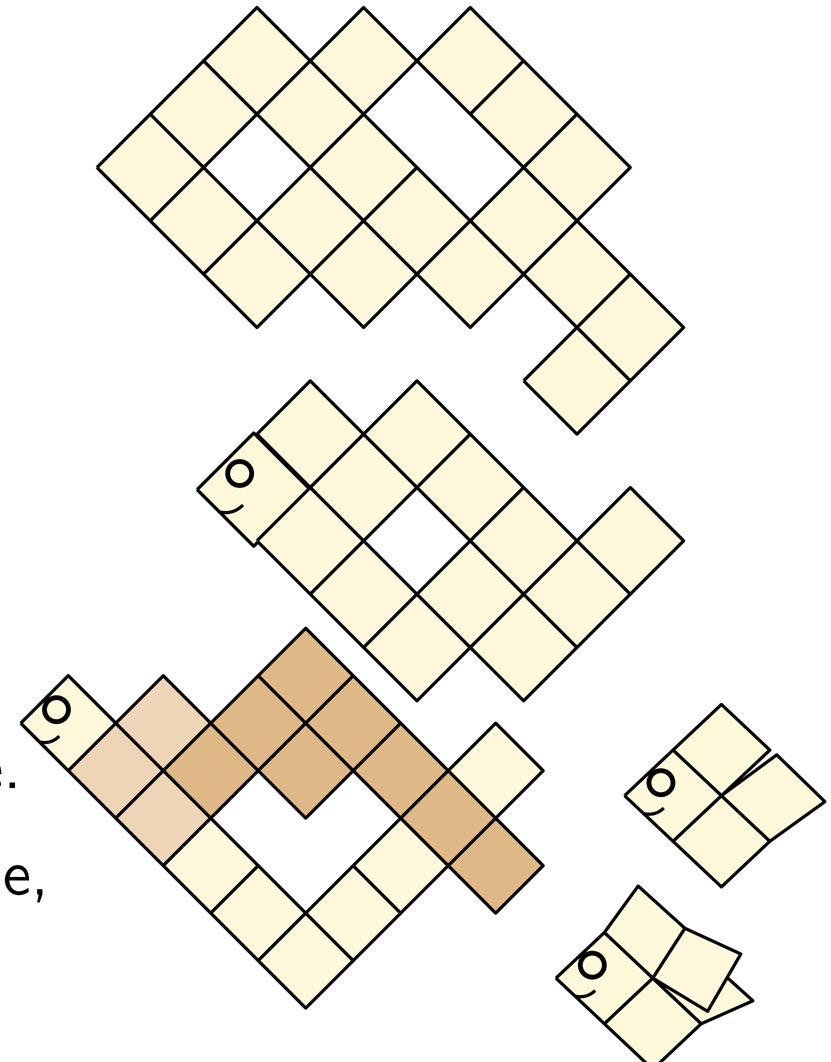
Fighting fish versus polyominoes

Polyomino = edge-connected set of cells **of the planar square lattice**

Directed polyominoes: there is a cell, the head, from which all cells can be reached by a left-to-right path.

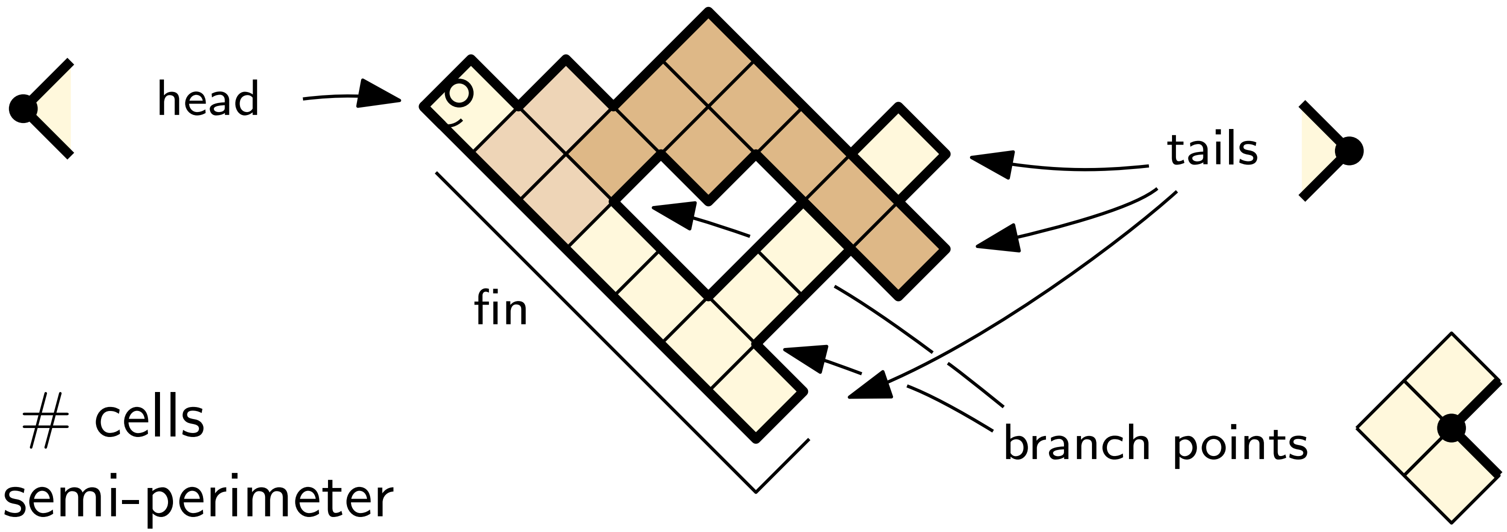
Proposition.
A fighting fish is a directed polyomino **iff** its projection in the plane is injective.

$\Rightarrow$ fighting fish do not all fit in the plane, *ie* they are not all polyominoes.



Conversely there are directed polyominoes that are not fighting fish:
Fighting fish are a generalization of directed polyominoes without holes.

Parameters of fighting fish



Area $=$ # cells

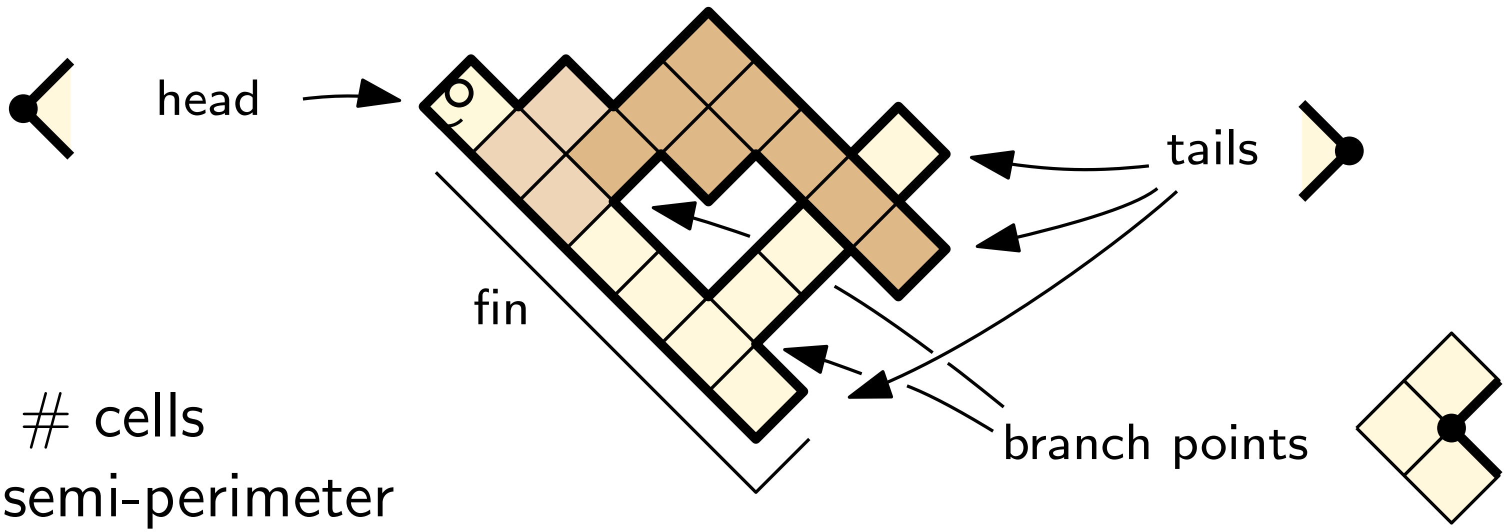
Size $=$ semi-perimeter

$= \#\{$upper free edges$\}$

$= \#\{$upper left free edges$\} + \#\{$upper right free edges$\}$

The fin length $= \#\{$ lower free edges from head to first tail $\}$

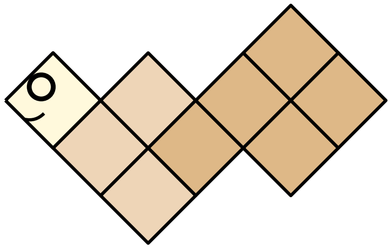
Parameters of fighting fish



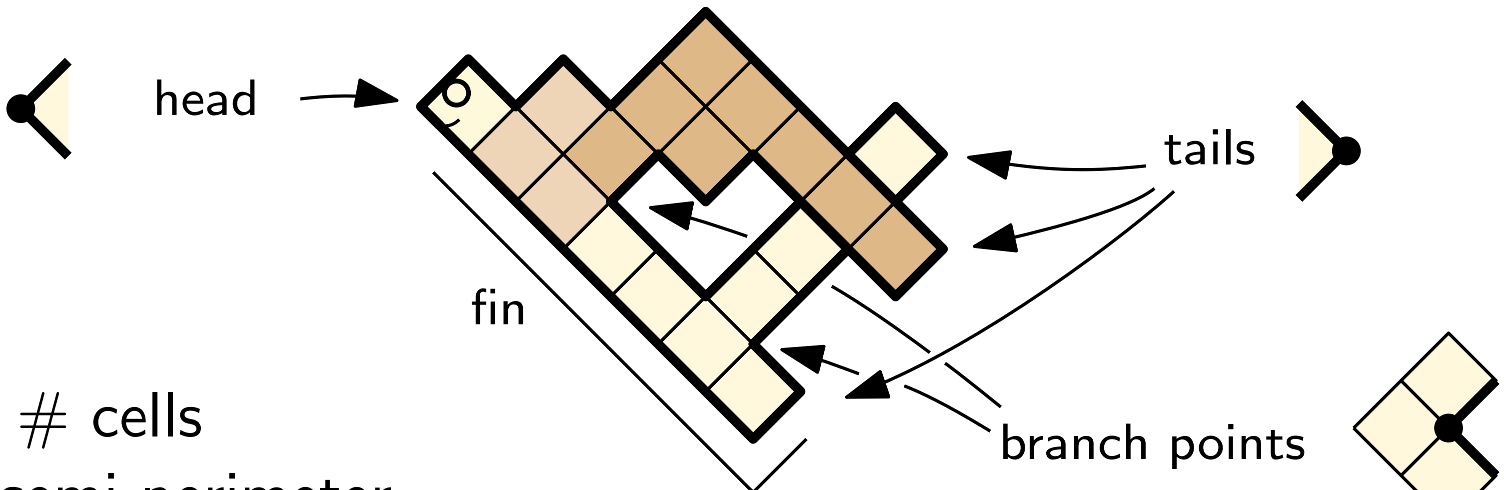
Area $=$ # cells
Size $=$ semi-perimeter
$\quad = \#\{$upper free edges$\}$
$\quad = \#\{$upper left free edges$\} + \#\{$upper right free edges$\}$

The fin length $= \#\{$ lower free edges from head to first tail $\}$

Fighting fish with exactly 1 tail



Parameters of fighting fish



Area $=$ # cells

Size $=$ semi-perimeter

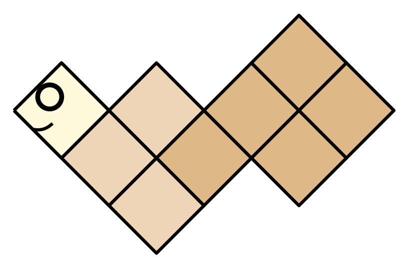
$\quad = \#\{$upper free edges$\}$

$\quad = \#\{$upper left free edges$\} + \#\{$upper right free edges$\}$

The fin length $= \#\{$ lower free edges from head to first tail $\}$

Fighting fish with exactly 1 tail

$=$ parallelogram polyominoes
aka staircase polygons



in this case, fin length $=$ semi-perimeter

Enumerative results

Enumerative results

fighting fish with 1 tail

Theorem (folklore)

$$\#\left\{\begin{array}{c}\text{parallelogram polyominos}\\ \text{with semi-perimeter } n+1\end{array}\right\} = \frac{1}{2n+1}\binom{2n}{n}$$

$$\#\left\{\begin{array}{c}\text{parallelogram polyominos with}\\ i \text{ top left and } j \text{ top right edges}\end{array}\right\} = \frac{1}{i+j-1}\binom{i+j-1}{i}\binom{i+j-1}{j}$$

Enumerative results

fighting fish with 1 tail

Theorem (folklore)

$$\#\left\{\begin{array}{c}\text{parallelogram polyominos}\\ \text{with semi-perimeter } n+1\end{array}\right\} = \frac{1}{2n+1}\binom{2n}{n}$$

$$\#\left\{\begin{array}{c}\text{parallelogram polyominos with}\\ i \text{ top left and } j \text{ top right edges}\end{array}\right\} = \frac{1}{i+j-1}\binom{i+j-1}{i}\binom{i+j-1}{j}$$

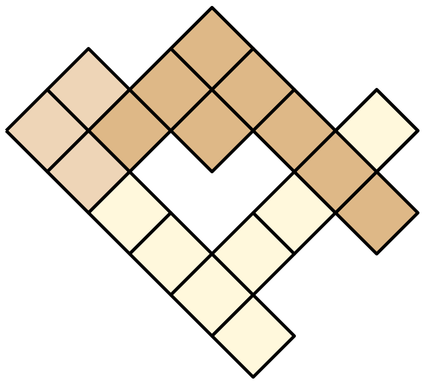
Theorem (D., Guerrini, Rinaldi, Schaeffer, 2016)

$$\#\left\{\begin{array}{c}\text{fighting fish}\\ \text{with semi-perimeter } n+1\end{array}\right\} = \frac{2}{(n+1)(2n+1)}\binom{3n}{n}$$

$$\#\left\{\begin{array}{c}\text{fighting fish with}\\ i \text{ top left and } j \text{ top right edges}\end{array}\right\} = \frac{1}{(2i+j-1)(2j+i-1)}\binom{2i+j-1}{i}\binom{2j+i-1}{j}$$

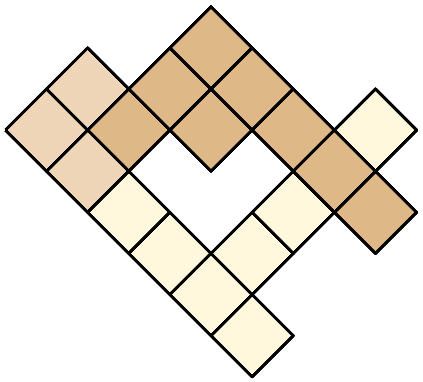
Fighting fish as random branching surfaces

Let F_n be a fighting fish taken uniformly at random among all fighting fish of size n.



Fighting fish as random branching surfaces

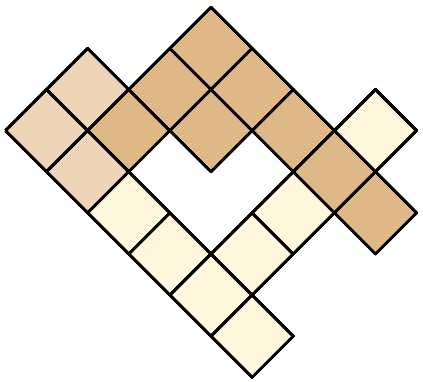
Let F_n be a fighting fish taken uniformly at random among all fighting fish of size n.



Theorem (D., Guerrini, Rinaldi, Schaeffer, *J. Physics A, 2016*)
The expected area of F_n is of order $n^{5/4}$

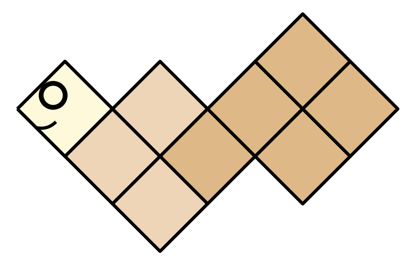
Fighting fish as random branching surfaces

Let F_n be a fighting fish taken uniformly at random among all fighting fish of size n.



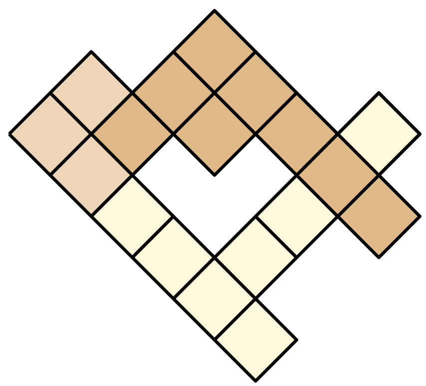
Theorem (D., Guerrini, Rinaldi, Schaeffer, *J. Physics A, 2016*)
The expected area of F_n is of order $n^{5/4}$

Compare to the known expected area $n^{3/2}$ of random parallelogram polyominoes of size n



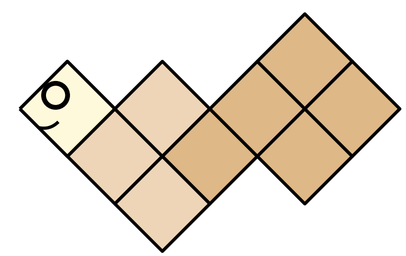
Fighting fish as random branching surfaces

Let F_n be a fighting fish taken uniformly at random among all fighting fish of size n.



Theorem (D., Guerrini, Rinaldi, Schaeffer, *J. Physics A, 2016*)
The expected area of F_n is of order $n^{5/4}$

Compare to the known expected area $n^{3/2}$ of random parallelogram polyominoes of size n



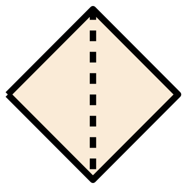
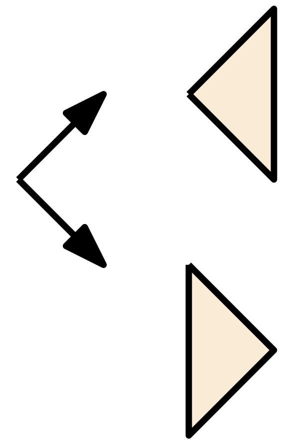
Uniform random fighting fish of size n gives a new model of random branching surfaces with original features.

Fish tails

We start by giving the definition of a slightly more general class: **Fighting fish tails.**

Fish tails

We start by giving the definition of a slightly more general class: **Fighting fish tails.**

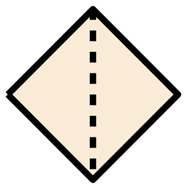
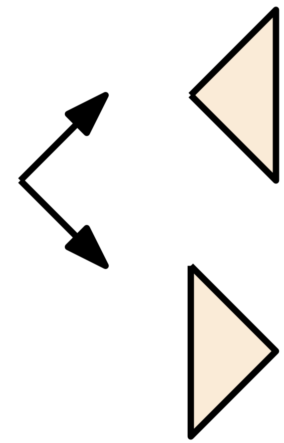
A cell  is made up by two scales 

Left scale

Right scale

Fish tails

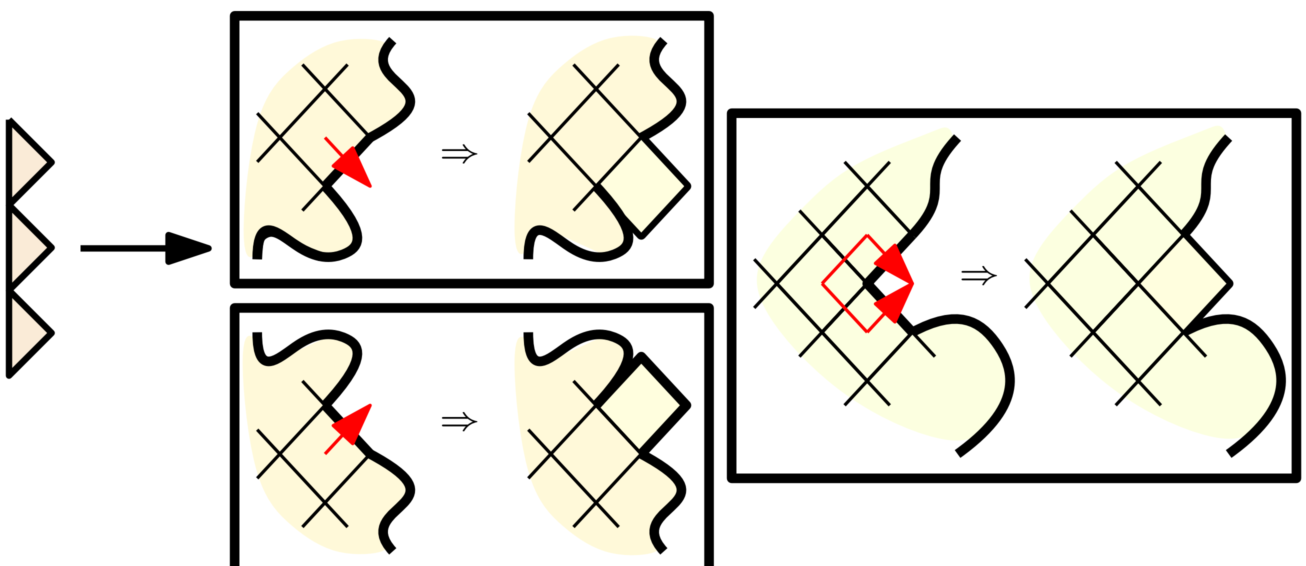
We start by giving the definition of a slightly more general class: **Fighting fish tails.**

A cell  is made up by two scales 

Left scale

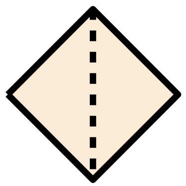
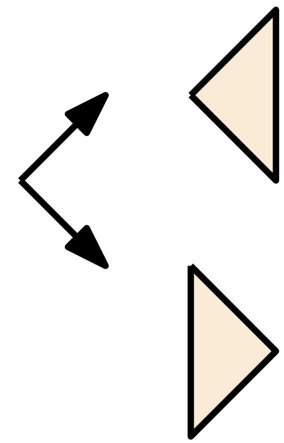
Right scale

A **fish tail** is a surface that can be obtained from a strip of right scales by a sequence of directed cell agregations.



Fish tails

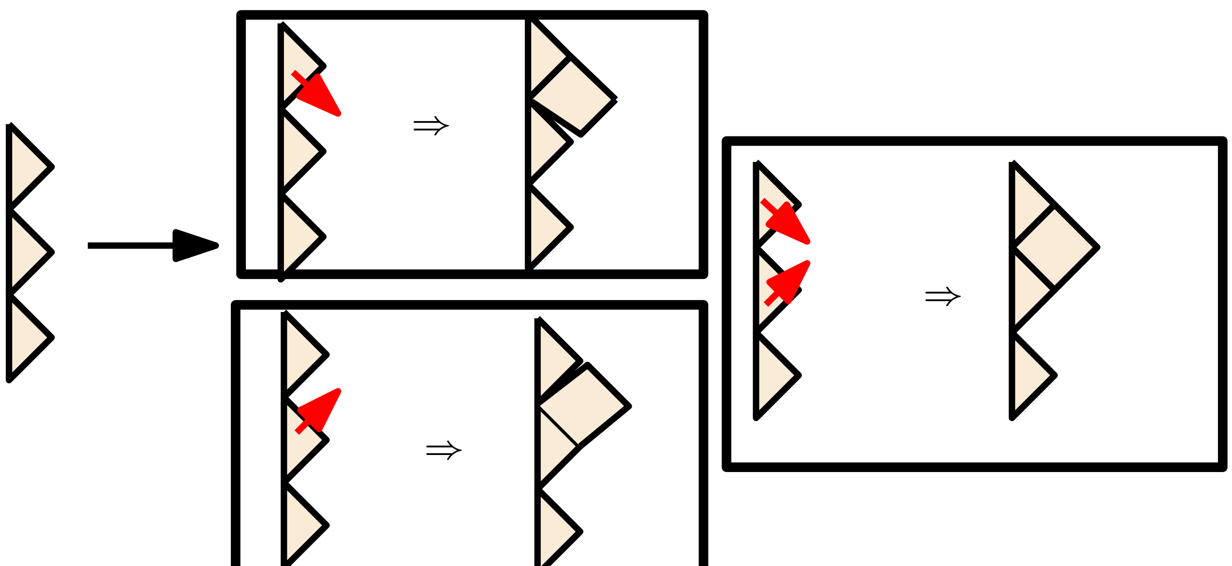
We start by giving the definition of a slightly more general class: **Fighting fish tails.**

A cell  is made up by two scales 

Left scale

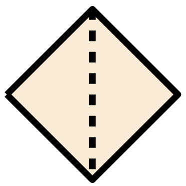
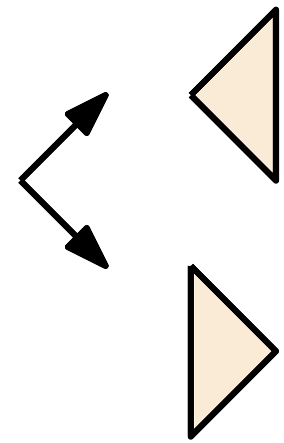
Right scale

A **fish tail** is a surface that can be obtained from a strip of right scales by a sequence of directed cell agregations.



Fish tails

We start by giving the definition of a slightly more general class: **Fighting fish tails.**

A cell  is made up by two scales 

Left scale

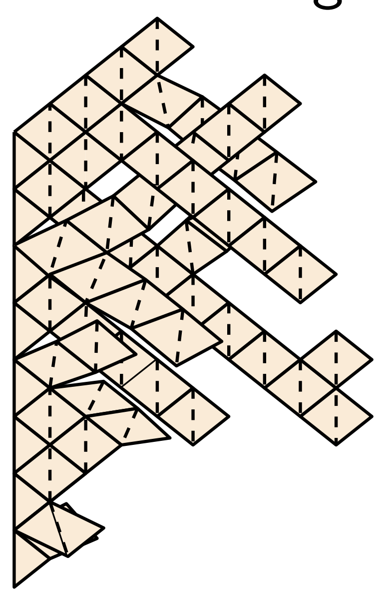
Right scale

A **fish tail** is a surface that can be obtained from a strip of right scales by a sequence of directed cell agregations.

area= the number of left and right scales in the fish tail

height= the number of right scales in the strip

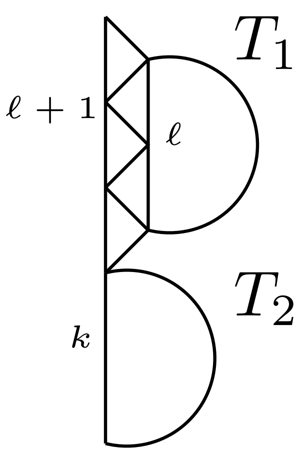
size= the number of upper left and right free edges



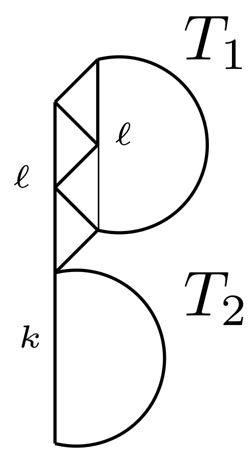
A recursive decomposition

A recursive decomposition

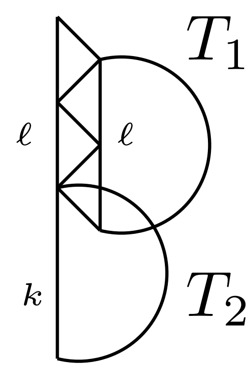
The empty fish is the unique fish tail with height 0.



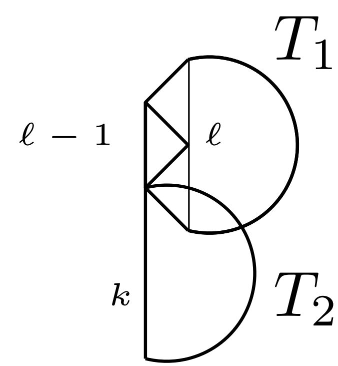
Operation u



Operation h



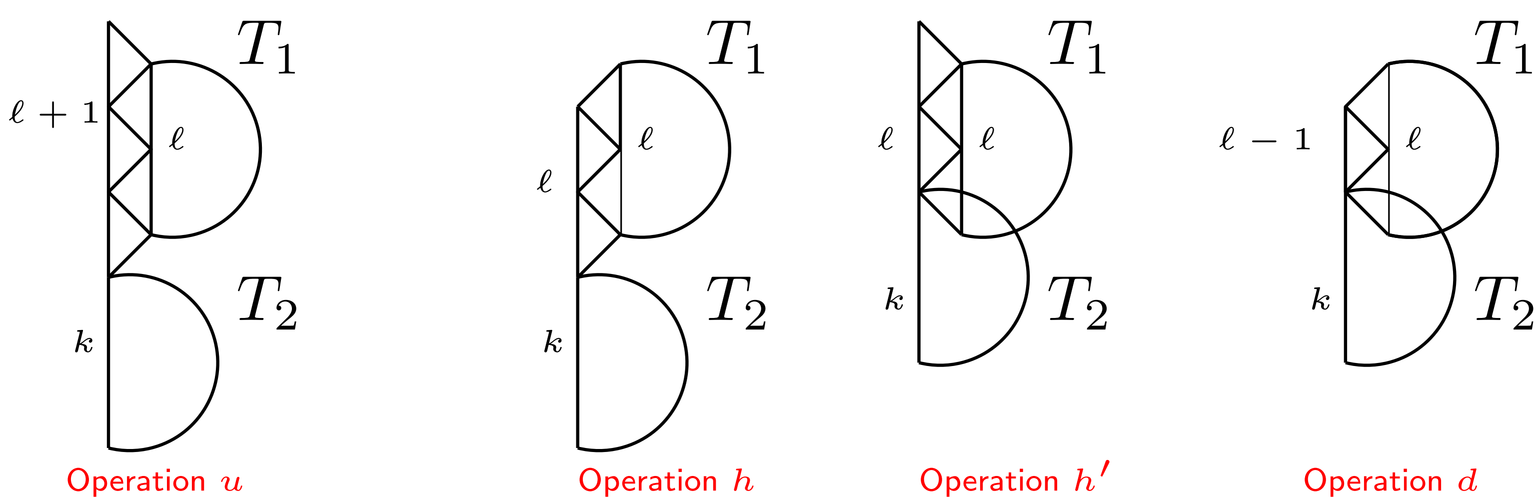
Operation h'



Operation d

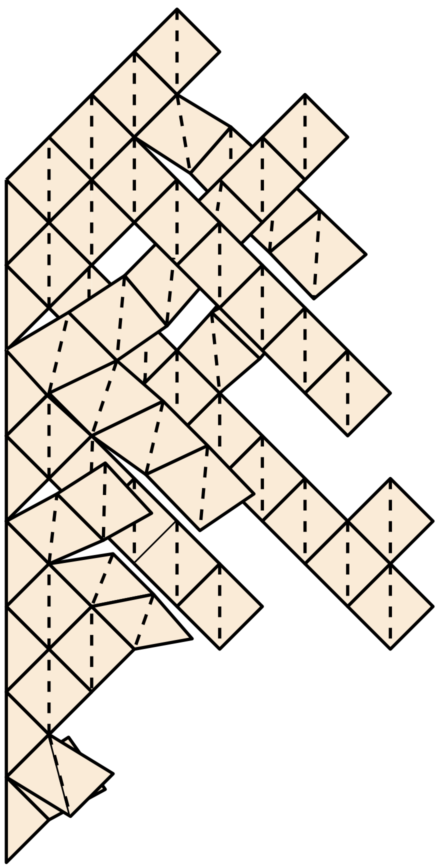
A recursive decomposition

The empty fish is the unique fish tail with height 0.

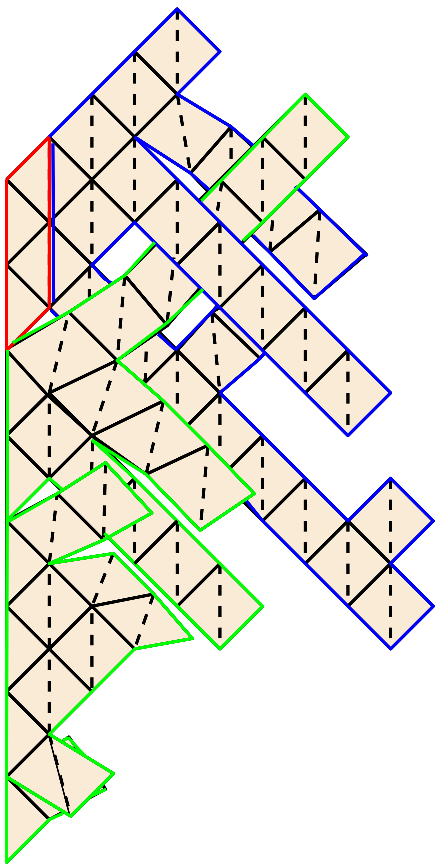


Every fish tail can be obtained in a unique way using operations u, h, h', d.

A recursive definition

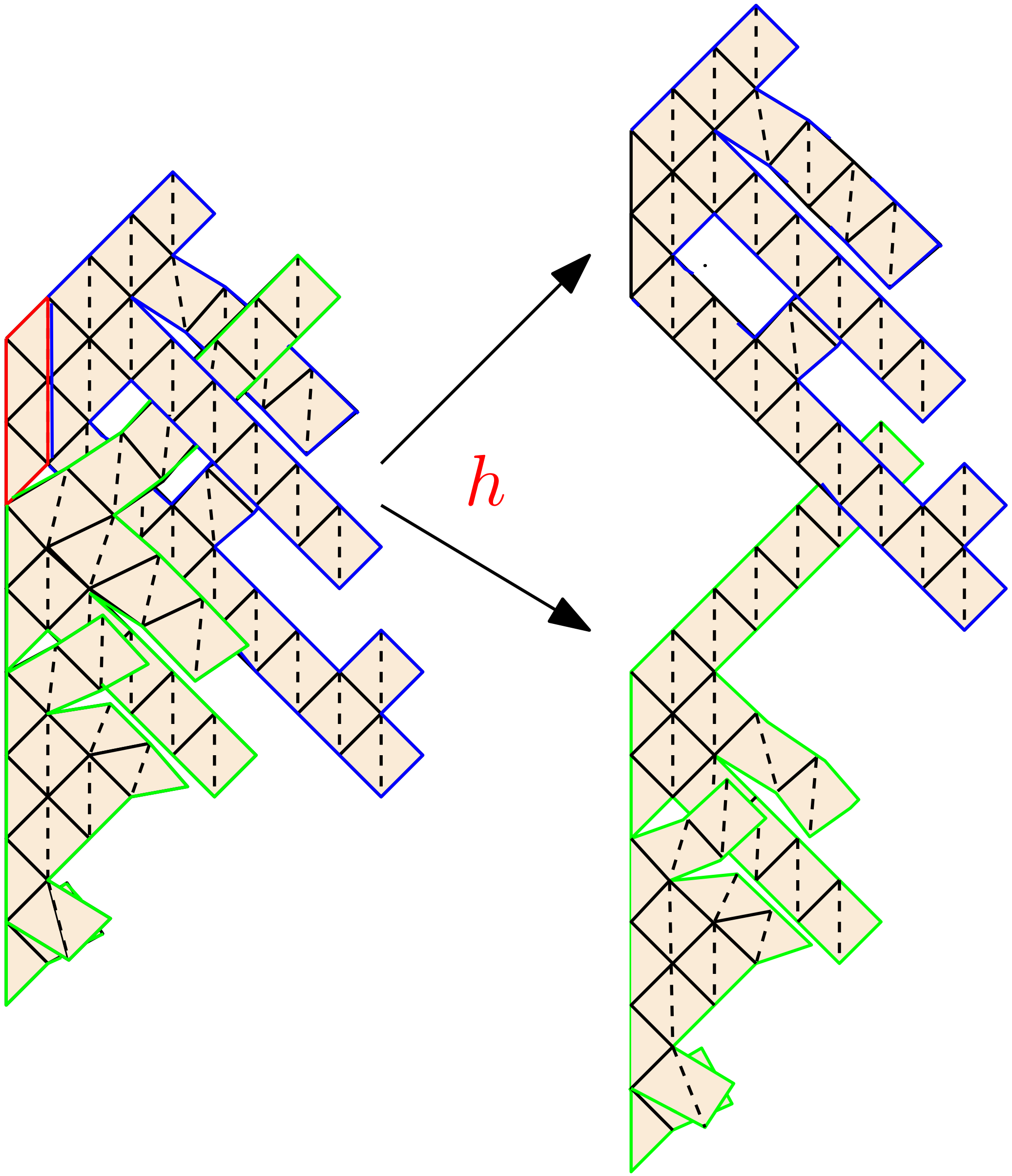


A recursive definition

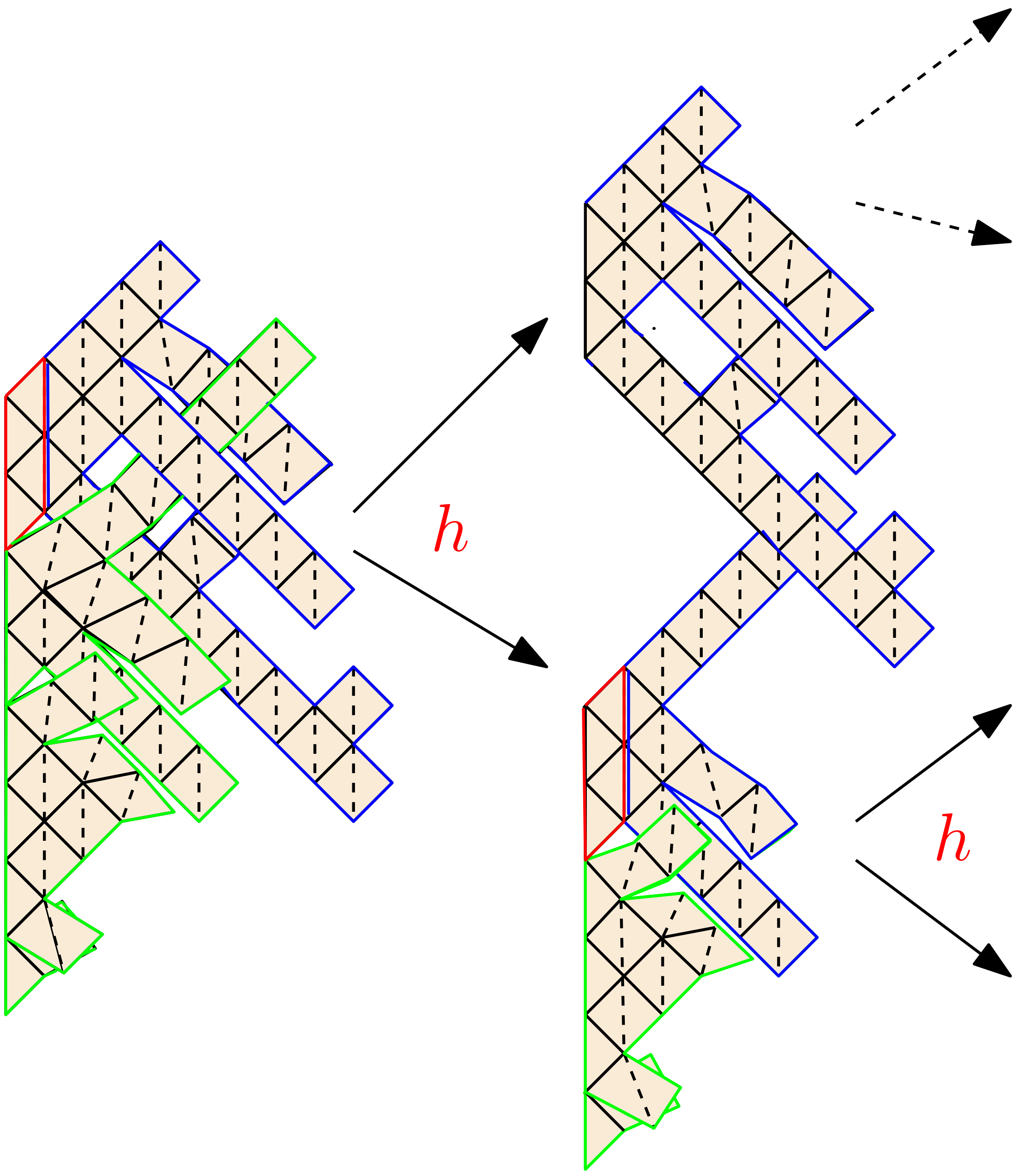


h

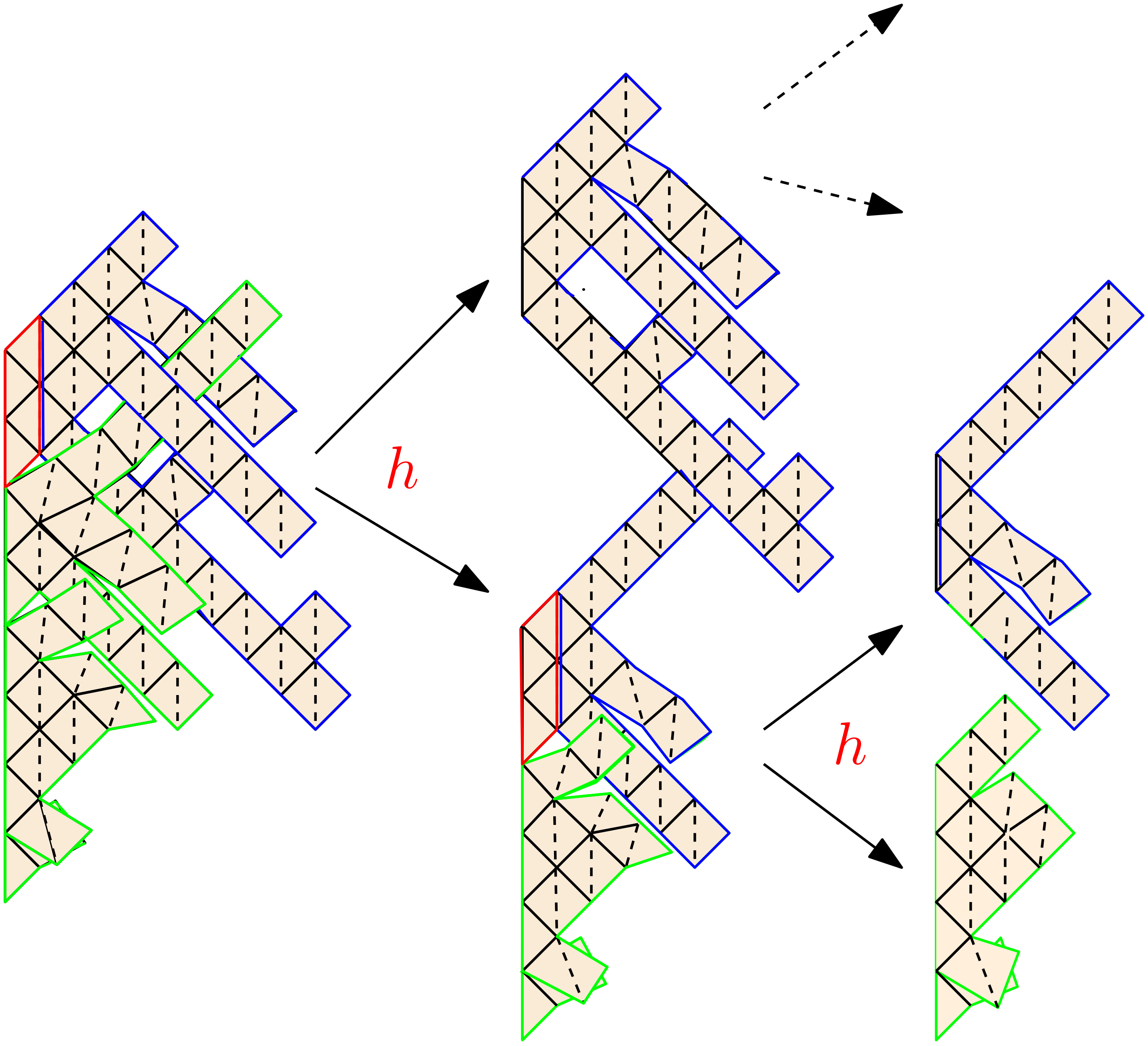
A recursive definition



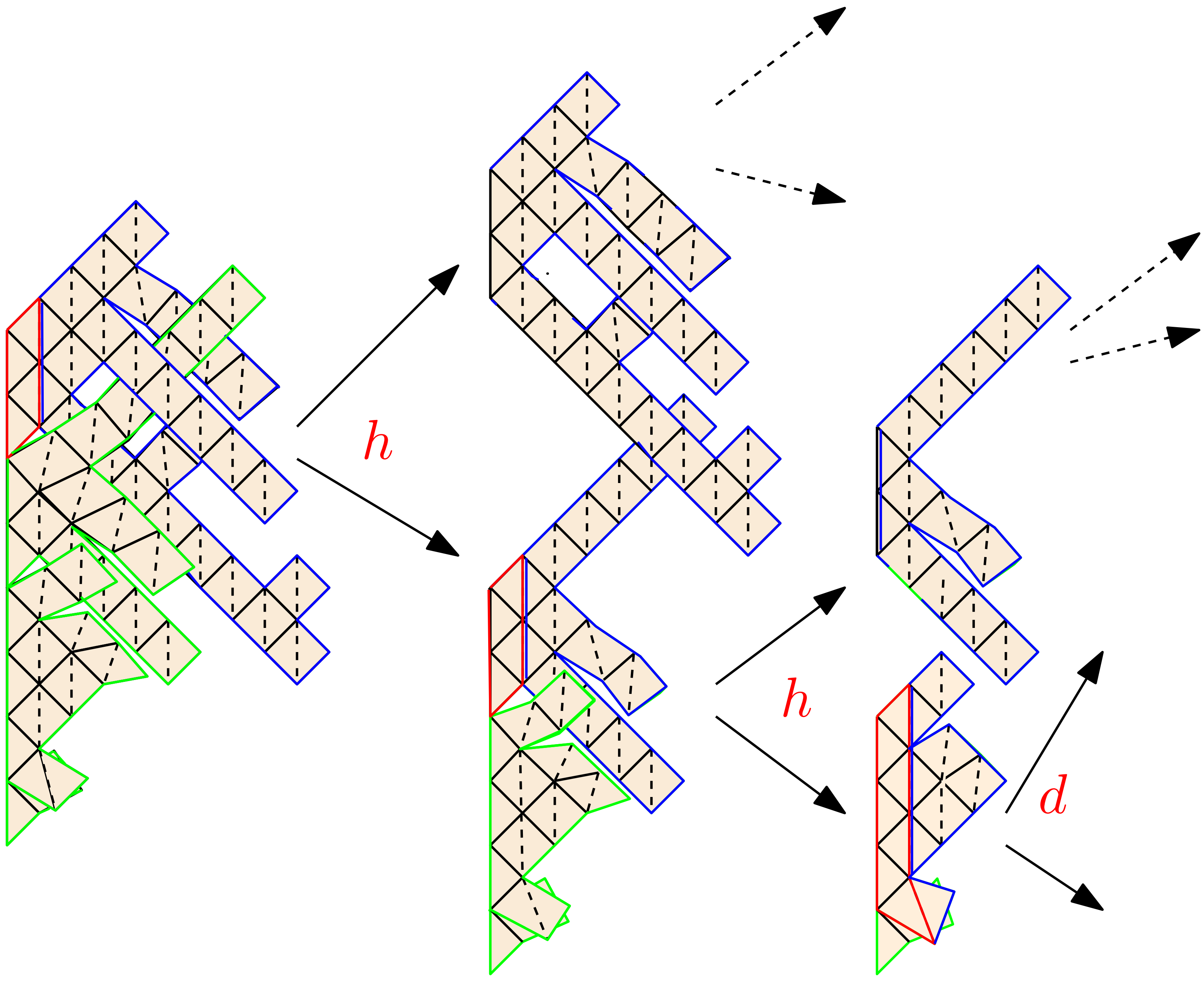
A recursive definition



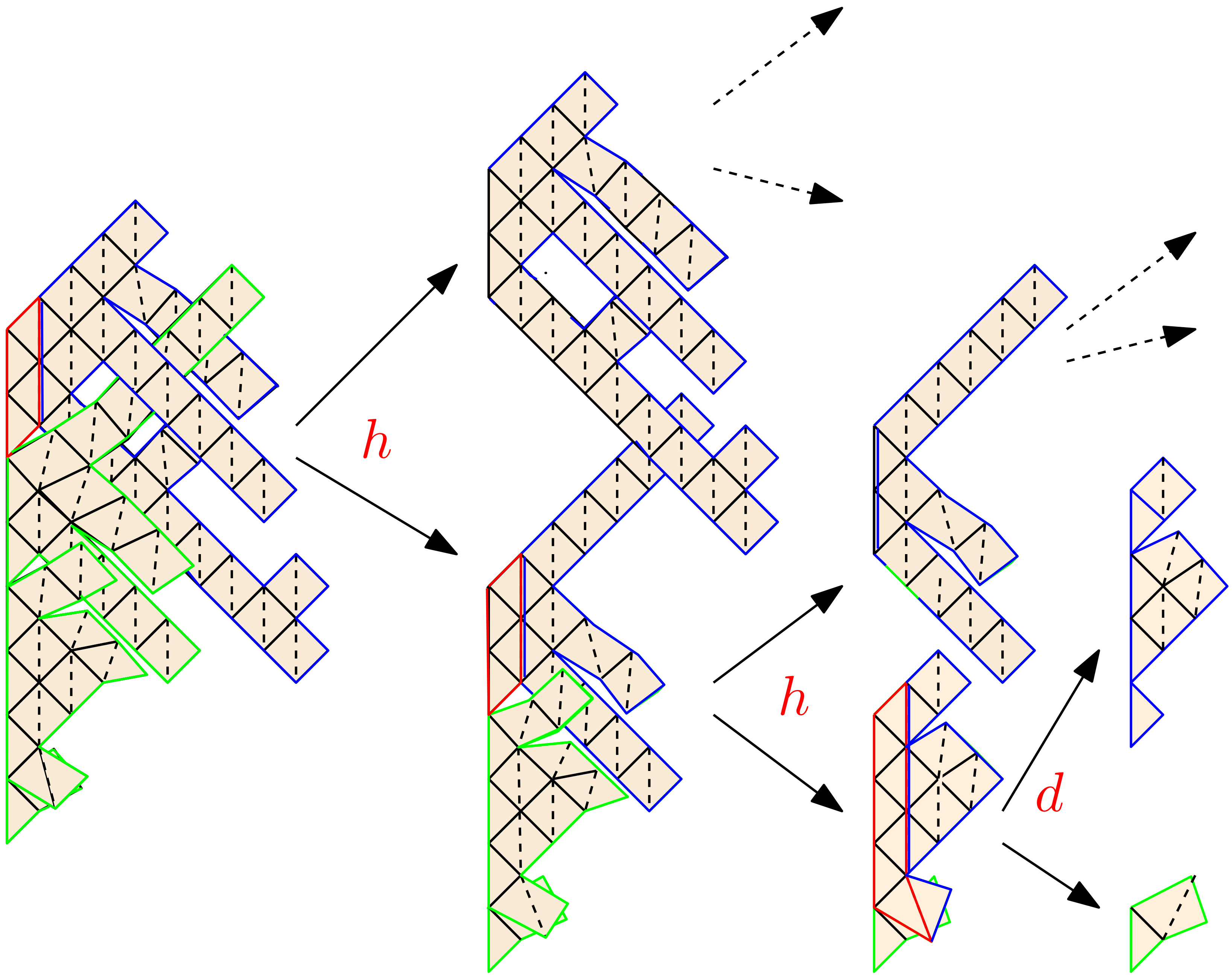
A recursive definition



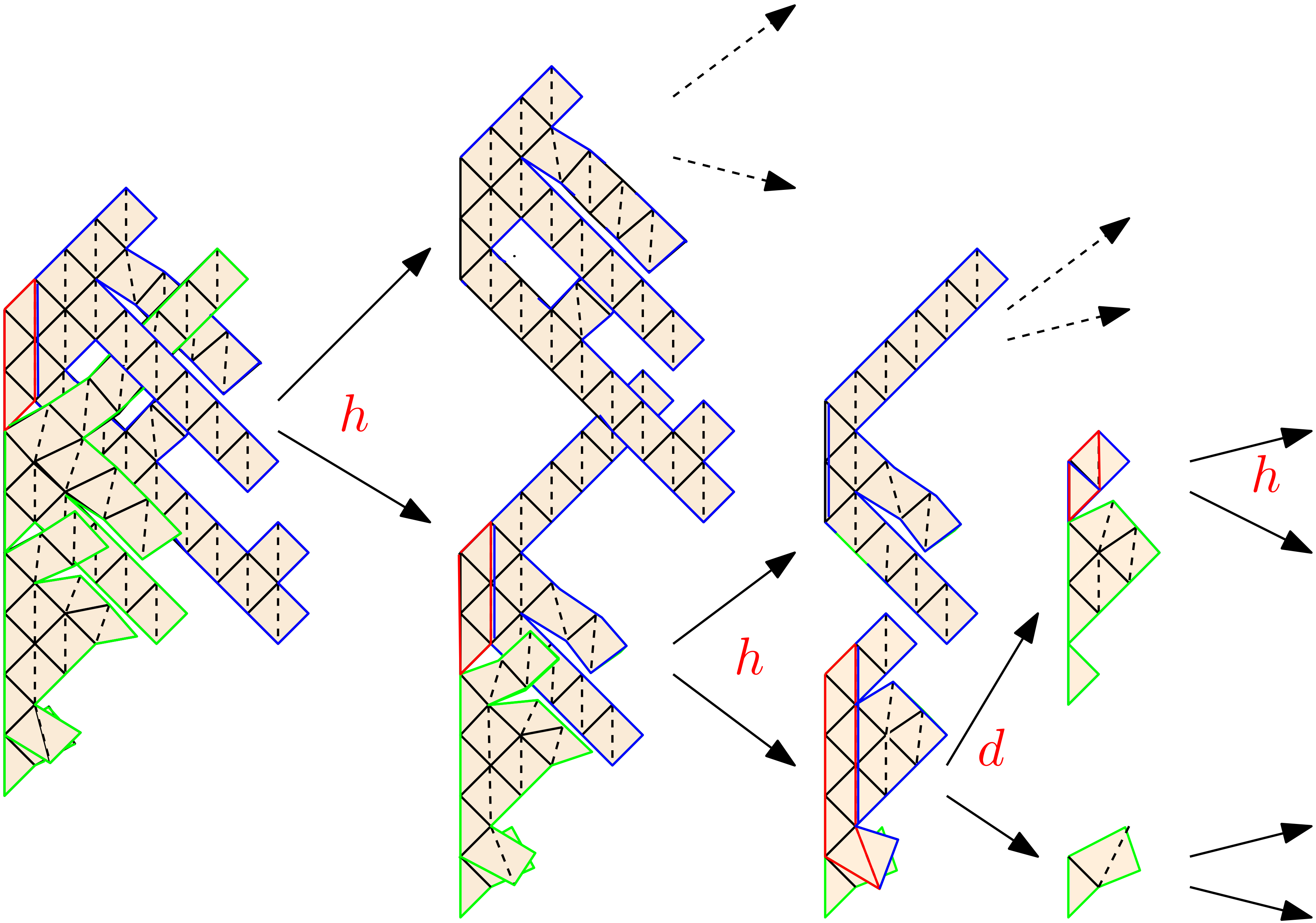
A recursive definition



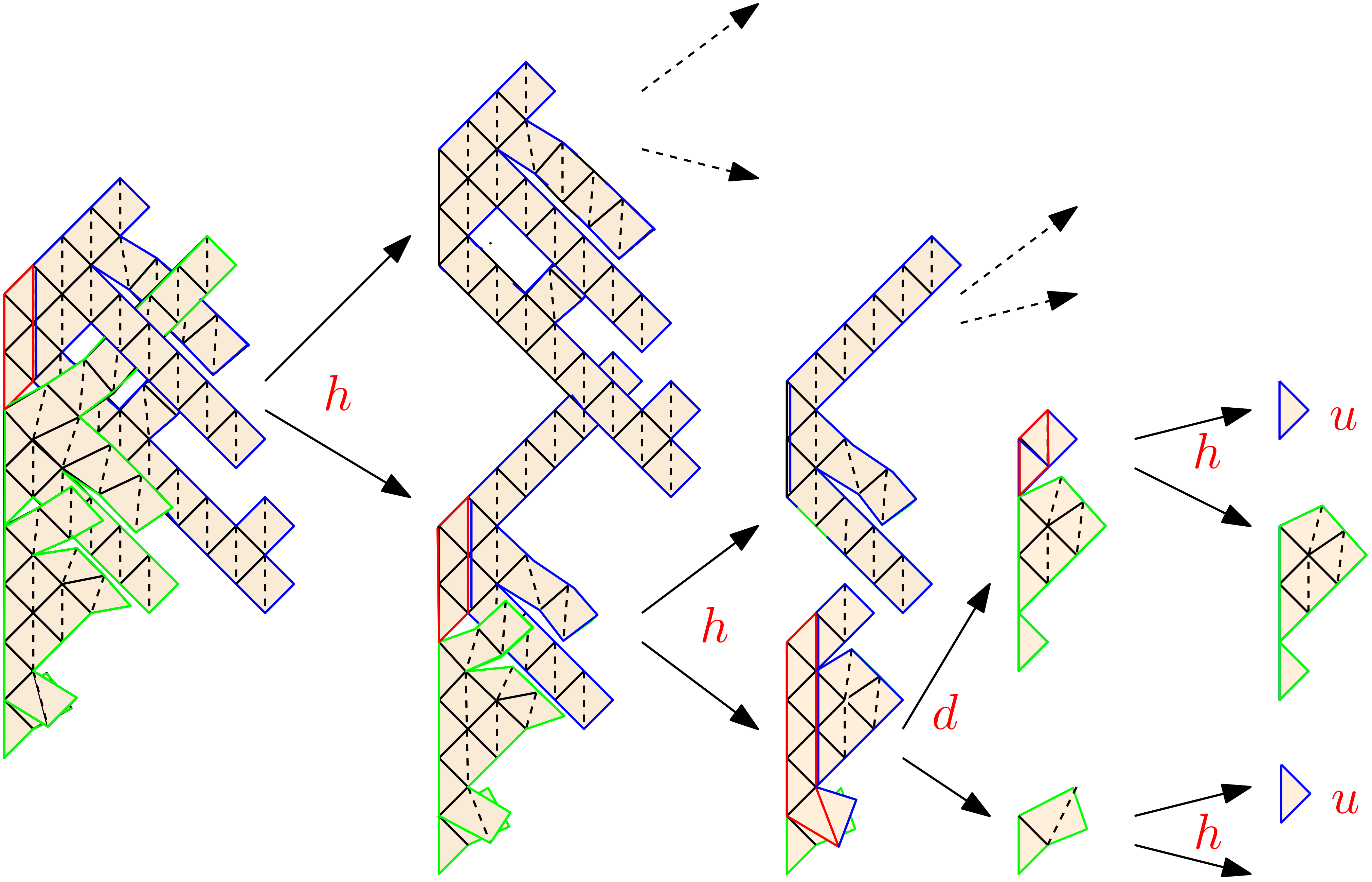
A recursive definition



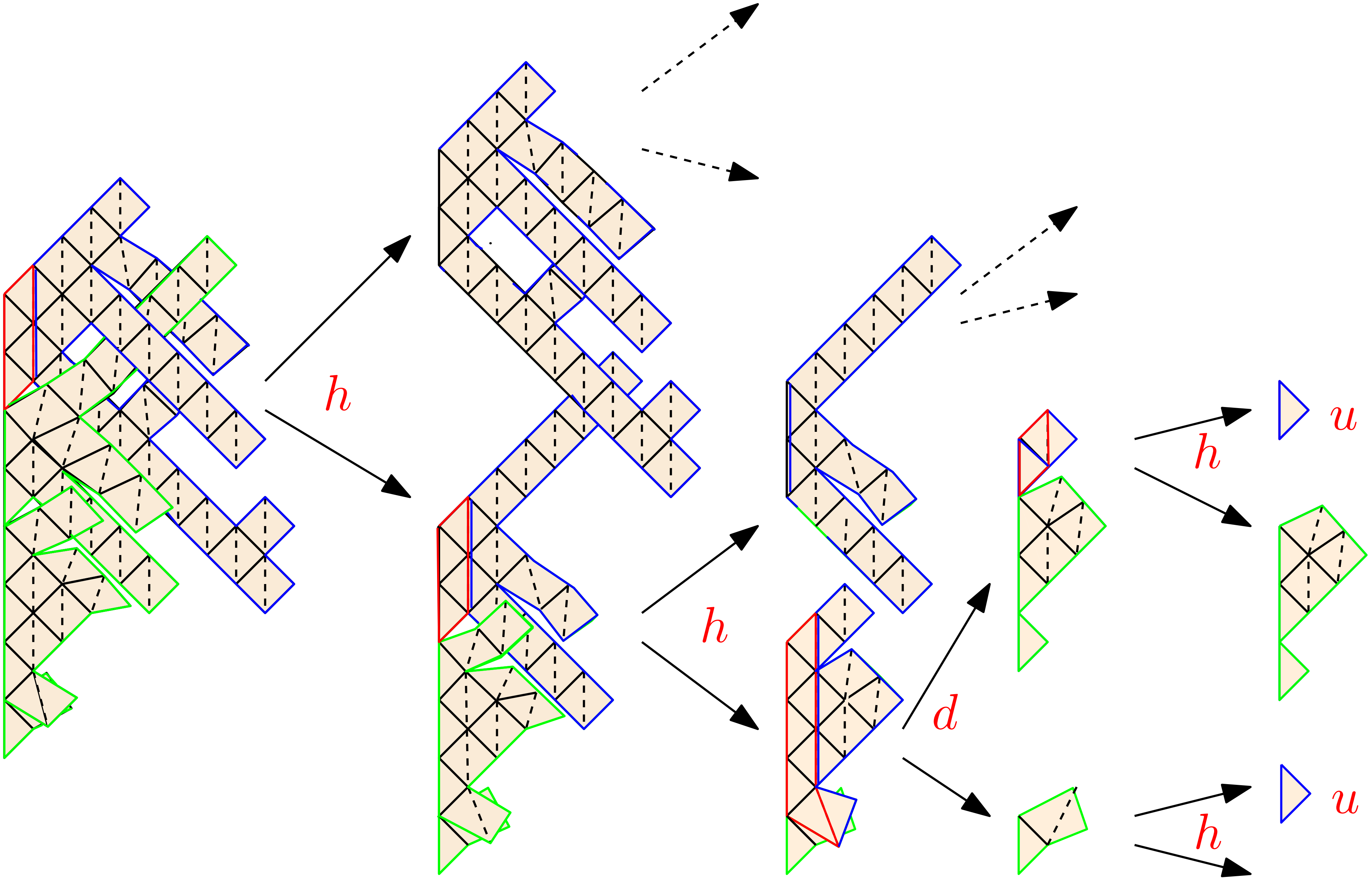
A recursive definition



A recursive definition



A recursive definition

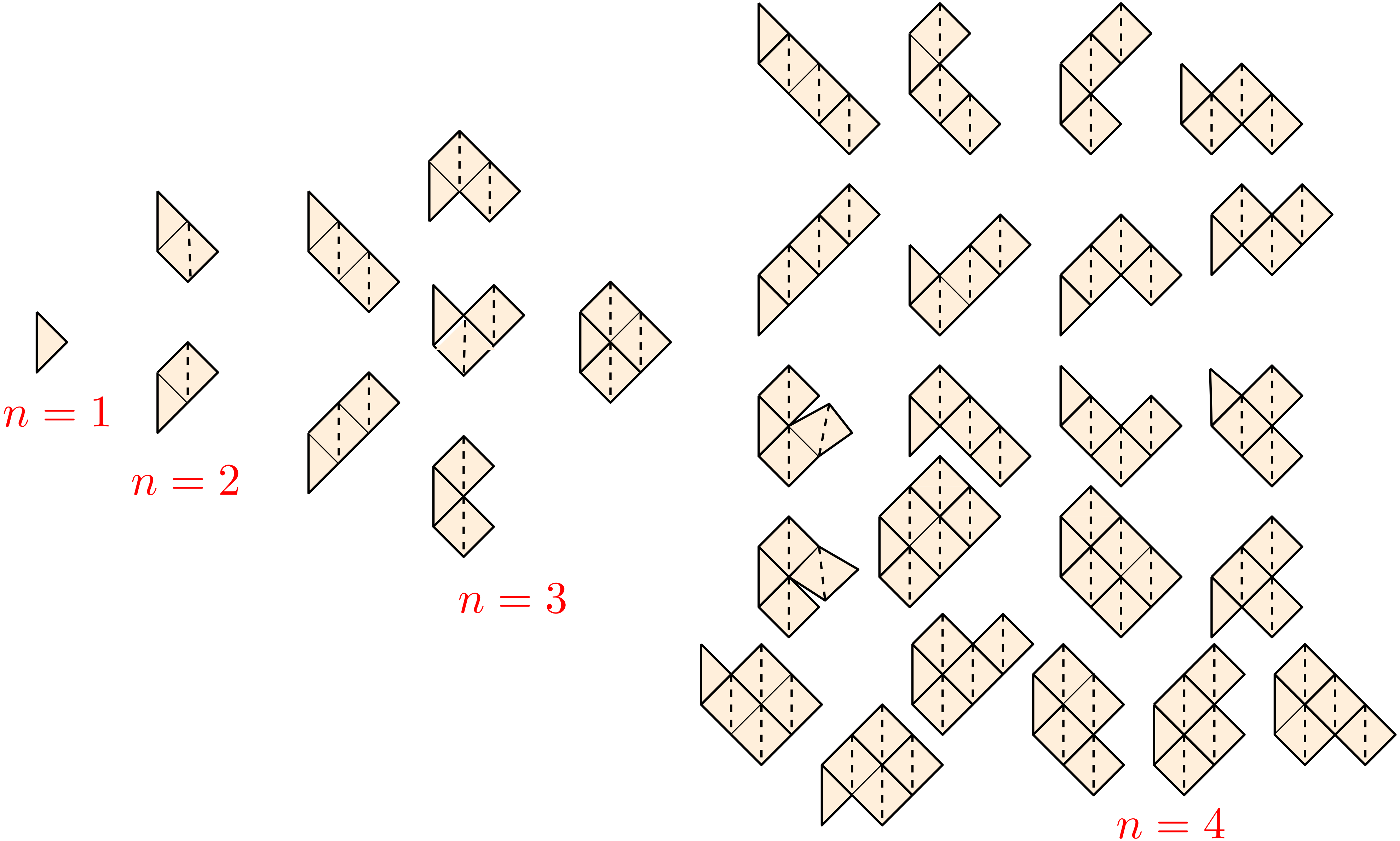


Fish tails vs fighting fish

Fish tails with height 1 and n free upper edges are in one-to-one correspondence with fighting fish with $n+1$ free upper edges.

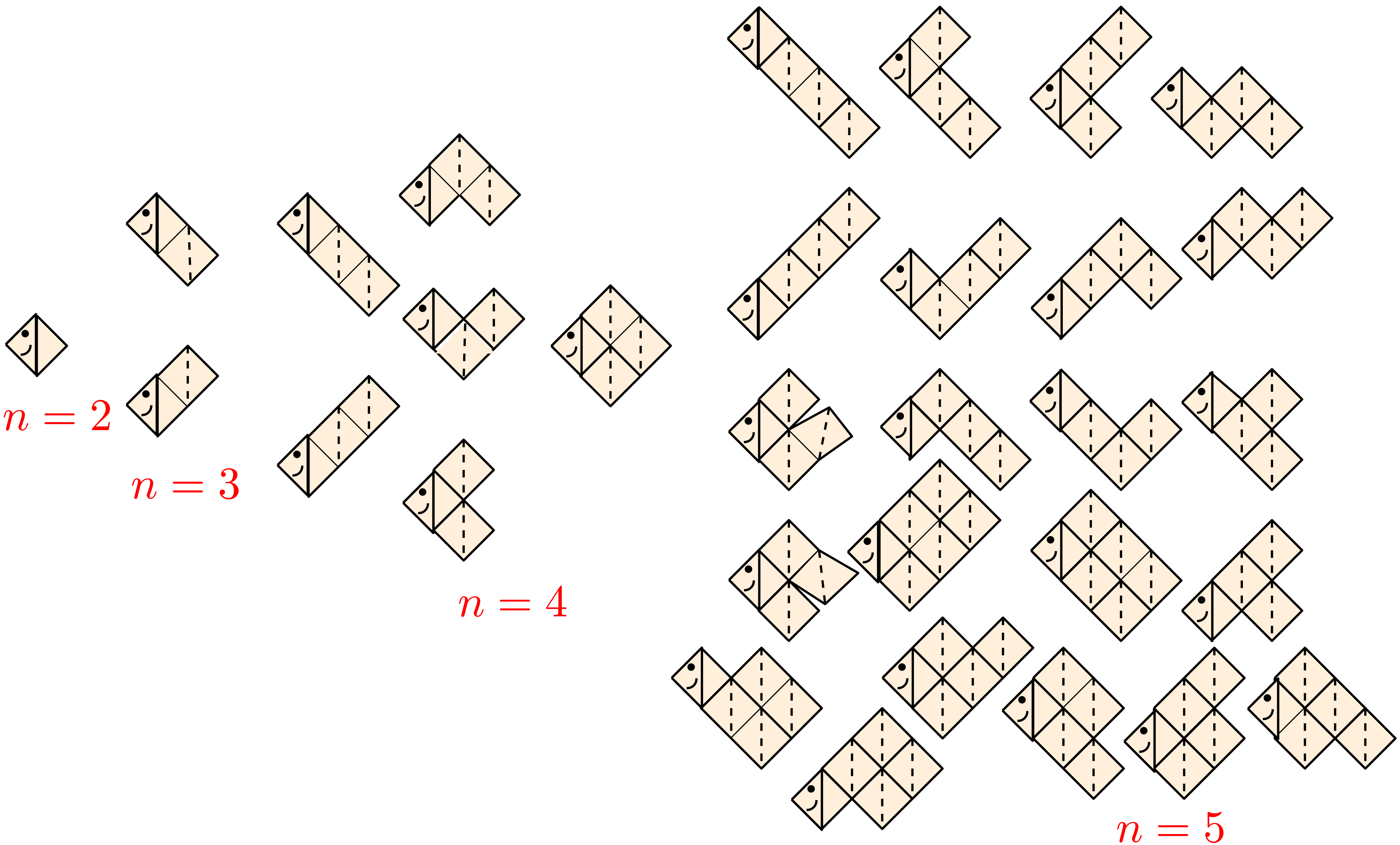
Fish tails vs fighting fish

Fish tails with height 1 and n free upper edges are in one-to-one correspondence with fighting fish with $n+1$ free upper edges.



Fish tails vs fighting fish

Fish tails with height 1 and n free upper edges are in one-to-one correspondence with fighting fish with $n+1$ free upper edges.



The generating function

Let $\mathcal{FT}$ be the set of fish tails without the empty fish tail.

Then $T(v,q,x,t) = \sum_{T \in \mathcal{FT}} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$ denote the generating function of fish tails, where

$h(T)$ is the height of T

$a(T)$ is the area of T

$c(T)$ is the number of tails of T

$n(T)$ is the semi-perimeter of T

The generating function

Let $\mathcal{FT}$ be the set of fish tails without the empty fish tail.

Then $T(v, q, x, t) = \sum_{T \in \mathcal{FT}} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$ denote the generating function of fish tails, where

$h(T)$ is the height of T

$a(T)$ is the area of T

$c(T)$ is the number of tails of T

$n(T)$ is the semi-perimeter of T

Let us denote

$$T(v, q) \equiv T(v, q, x, t) \qquad\qquad f(q) = [v]T(v, q)$$

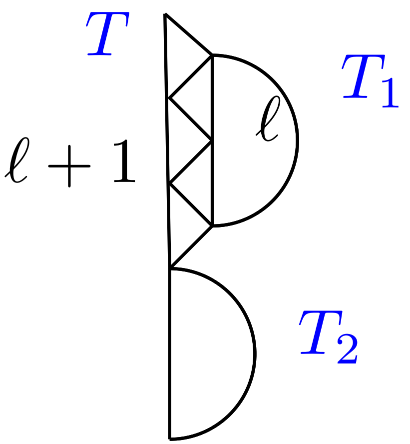
We are going to write the functional equation associated with the previous construction.

The functional equation for fish tails

$T(v, q, x, t) = \sum_{T \in \mathcal{FT}} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$

The functional equation for fish tails

$$T(v,q,x,t) = \sum_{T \in \mathcal{FT}} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$$



Operation u

$$n(T) = n(T_1) + n(T_2) + 1$$

$$h(T) = h(T_1) + h(T_2) + 1$$

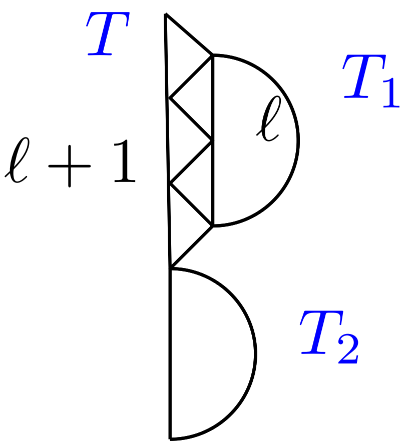
$$c(T) = c(T_1) + c(T_2) \text{ if } \ell := h(T_1) \neq 0$$

$$c(T) = c(T_2) + 1 \quad \text{if } \ell = 0$$

$$a(T) = a(T_1) + a(T_2) + 2\ell + 1$$

The functional equation for fish tails

$T(v,q,x,t) = \sum_{T\in\mathcal{FT}} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$



Operation u

$n(T) = n(T_1) + n(T_2) + 1$

$h(T) = h(T_1) + h(T_2) + 1$

$c(T) = c(T_1) + c(T_2)$ if $\ell := h(T_1) \neq 0$

$c(T) = c(T_2) + 1$ if $\ell = 0$

$a(T) = a(T_1) + a(T_2) + 2\ell + 1$

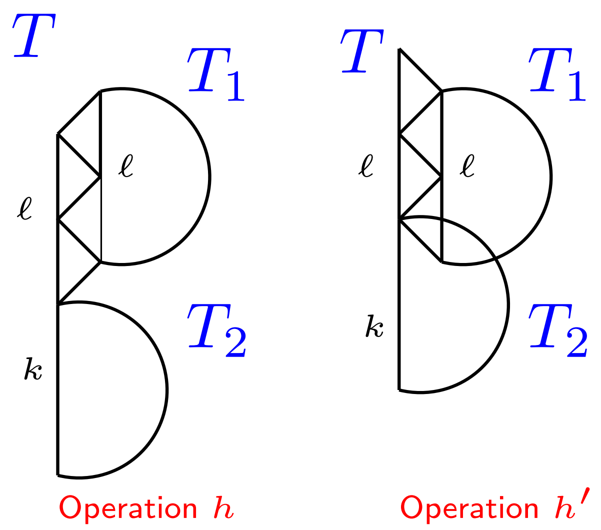
Operation u gives the term

$$tvqT(vq^2,q,x,t)(T(v,q,x,t)+1) + tvqx(T(v,q,x,t)+1)$$

$$= tvq(T(vq^2,q)+x)(T(v,q)+1)$$

The functional equation for fish tails

$$T(v,q,x,t) = \sum_{T\in\mathcal{FT}} v^{h(T)}q^{a(T)}x^{c(T)}t^{n(T)}$$



$$n(T) = n(T_1) + n(T_2) + 1$$

$$h(T) = h(T_1) + h(T_2)$$

$$c(T) = c(T_1) + c(T_2) \;\; (\ell \neq 0)$$

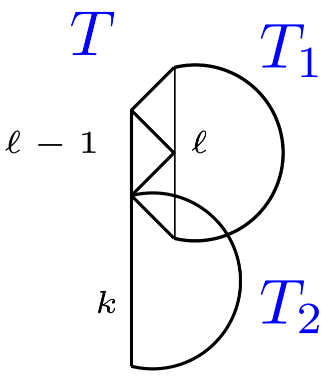
$$a(T) = a(T_1) + a(T_2) + 2\ell$$

Operation h and h' give the term

$$2t(T(vq^2,q)(T(v,q)+1)$$

The functional equation for fish tails

$$T(v,q,x,t) = \sum_{T\in\mathcal{FT}} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$$



Operation d

$$n(T) = n(T_1) + n(T_2) + 1$$

$$h(T) = h(T_1) + h(T_2) - 1 \qquad (\ell > 1)$$

$$c(T) = c(T_1) + c(T_2)$$

$$a(T) = a(T_1) + a(T_2) + 2\ell - 1$$

Operation u gives the term

$$\frac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

Enumeration wrt the perimeter and number of tails

The functional equation

$$\begin{aligned} T(v,q) = \quad & tvq(T(vq^2,q)+x)(T(v,q)+1) + 2tT(vq^2,q)(T(v,q)+1) \\ & + \tfrac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1) \end{aligned}$$

Enumeration wrt the perimeter and number of tails

The functional equation

$$T(v,q) = \quad tvq(T(vq^2,q)+x)(T(v,q)+1) + 2tT(vq^2,q)(T(v,q)+1)$$

$$+\tfrac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

Letting $q = 1$ the master equation reduces to

$$T(v) = \quad tv(T(v)+x)(T(v)+1) + 2tT(v)(T(v)+1)$$

$$+\tfrac{t}{v}(T(v) - vf)(T(v)+1)$$

where $T(v) \equiv T(v,1)$ and $f \equiv f(1)$

Enumeration wrt the perimeter and number of tails

The functional equation

$$T(v,q) = \quad tvq(T(vq^2,q)+x)(T(v,q)+1) + 2tT(vq^2,q)(T(v,q)+1)$$
$$+\tfrac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

Letting $q=1$ the master equation reduces to

$$T(v) = \quad tv(T(v)+x)(T(v)+1) + 2tT(v)(T(v)+1)$$
$$+\tfrac{t}{v}(T(v) - vf)(T(v)+1)$$

where $T(v) \equiv T(v,1)$ and $f \equiv f(1)$

This equation is now a polynomial equation with one catalytic variable and it admits an explicitly computable algebraic solution.
(Bousquet-Mélou and Jehanne, *J. Combin. Theory Ser.B*, 2006)

Enumeration wrt semi-perimeter and number of tails

$$
\begin{aligned}
T(v) = \quad & tv(T(v)+x)(T(v)+1) + 2tT(v)(T(v)+1) \\
& + \tfrac{t}{v}(T(v) - vf)(T(v)+1)
\end{aligned}
$$

We apply the Bousquet-Mélou Jehanne trick:

Enumeration wrt semi-perimeter and number of tails

$$\begin{aligned} T(v) = \quad & tv(T(v)+x)(T(v)+1) + 2tT(v)(T(v)+1) \\ & + \tfrac{t}{v}(T(v)-vf)(T(v)+1) \end{aligned}$$

We apply the Bousquet-Mélou Jehanne trick:

Upon deriving with respect to v we obtain the following equation

$$(1 - tv(T(v)+x) - 2tT(v) - \tfrac{t}{v}(T(v)-vf) - (tv+2t+\tfrac{t}{v})(T(v)+1))\tfrac{dT(v)}{dv} =$$

$$= (T(v)+1)(t(T(v)+x) - \tfrac{t}{v^2}(T(v)-vf) - \tfrac{t}{v}f)$$

Enumeration wrt semi-perimeter and number of tails

$$T(v) = \quad tv(T(v)+x)(T(v)+1) + 2tT(v)(T(v)+1)$$

$$+\tfrac{t}{v}(T(v)-vf)(T(v)+1)$$

We apply the Bousquet-Mélou Jehanne trick:

Upon deriving with respect to v we obtain the following equation

$$(1 - tv(T(v)+x) - 2tT(v) - \tfrac{t}{v}(T(v)-vf) - (tv+2t+\tfrac{t}{v})(T(v)+1))\tfrac{dT(v)}{dv} =$$

$$= (T(v)+1)(t(T(v)+x) - \tfrac{t}{v^2}(T(v)-vf) - \tfrac{t}{v}f)$$

There is a power series $V \equiv V(t)$, such that setting $v = V$ cancels the left hand side:

$$(1 - tV(T(V)+x) - 2tT(V) - \tfrac{t}{V}(T(V)-Vf) - (tV+2t+\tfrac{t}{V})(T(V)+1)) = 0$$

we then also have $$(T(V)+1)(t(T(V)+x) - \tfrac{t}{V^2}(T(V)-Vf) - \tfrac{t}{V}f) = 0$$

and the main equation gives a third equation.

Enumeration wrt semi-perimeter and number of tails

Simplifying the previous system of equations we obtain

$$\begin{aligned} V &= t(1+V+\tfrac{xV^2}{1-V})^2 \\ f &= xV - x^2\tfrac{V^3}{(1-V)^2} \\ T(V) &= \tfrac{xV^2}{1-V^2} \end{aligned}$$

Enumeration wrt semi-perimeter and number of tails

Simplifying the previous system of equations we obtain

$$\begin{aligned} V &= t(1 + V + \tfrac{xV^2}{1-V})^2 \\ f &= xV - x^2 \tfrac{V^3}{(1-V)^2} \\ T(V) &= \tfrac{xV^2}{1-V^2} \end{aligned}$$

By setting $x = 1$ we obtain equations for fighting fish according to the semi-perimeter only.

Enumeration wrt semi-perimeter and number of tails

Simplifying the previous system of equations we obtain

$$\begin{aligned} V &= t(1 + V + \tfrac{xV^2}{1-V})^2 \\ f &= xV - x^2 \tfrac{V^3}{(1-V)^2} \\ T(V) &= \tfrac{xV^2}{1-V^2} \end{aligned}$$

By setting $x = 1$ we obtain equations for fighting fish according to the semi-perimeter only.

$$\begin{aligned} V &= \tfrac{t}{(1-V)^2} \\ f &= V - \tfrac{V^3}{(1-V)^2} \end{aligned}$$

Enumeration wrt semi-perimeter and number of tails

Simplifying the previous system of equations we obtain

$$
\begin{aligned}
V &= t\left(1+V+\tfrac{xV^2}{1-V}\right)^2 \\
f &= xV - x^2\tfrac{V^3}{(1-V)^2} \\
T(V) &= \tfrac{xV^2}{1-V^2}
\end{aligned}
$$

By setting $x = 1$ we obtain equations for fighting fish according to the semi-perimeter only.

$$
\begin{aligned}
V &= \tfrac{t}{(1-V)^2} \\
f &= V - \tfrac{V^3}{(1-V)^2}
\end{aligned}
$$

Lagrange inversion formula

The number of fighting fish with size $n+1$

$$\longrightarrow [t^n]f = \frac{2}{(n+1)(2n+1)}\binom{3n}{n}$$

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$\begin{aligned} f &= xV - x^2\frac{V^3}{(1-V)^2} \\ V &= t(1+V+\frac{xV^2}{1-V})^2 \end{aligned}$$

$\longrightarrow$

$$[x]f = [x^0]V = V_0$$

where $V_0 = t(1+V_0)^2$

$$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$$

Parallelogram polyominoes of size $n+1$

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$f = xV - x^2\frac{V^3}{(1-V)^2}$$

$$V = t\left(1 + V + \frac{xV^2}{1-V}\right)^2$$

$\longrightarrow$

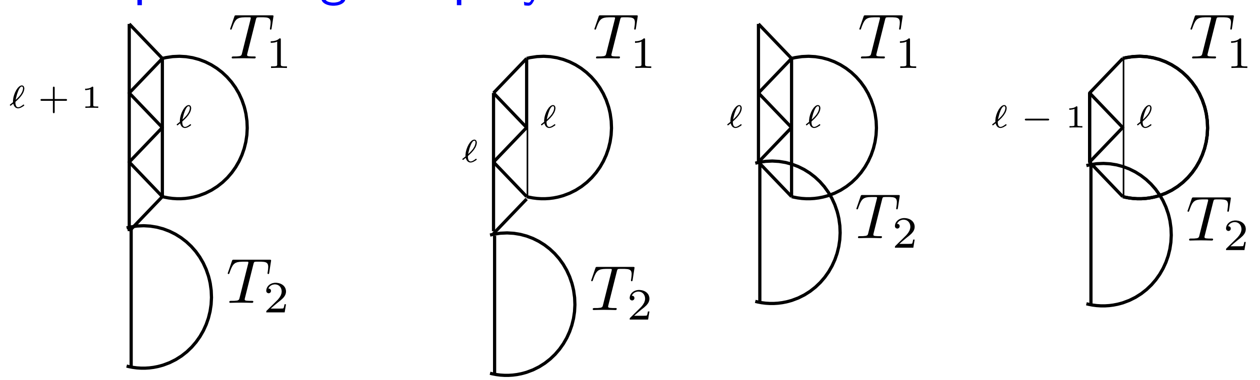
$$[x]f = [x^0]V = V_0$$

where $V_0 = t(1+V_0)^2$

$$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$$

Parallelogram polyominoes of size $n+1$

Our decomposition generalizes a Temperly like decomposition for parallelogram polyominoes.



Operation u Operation h Operation h' Operation d

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$f = xV - x^2\frac{V^3}{(1-V)^2}$$

$$V = t\left(1 + V + \frac{xV^2}{1-V}\right)^2$$

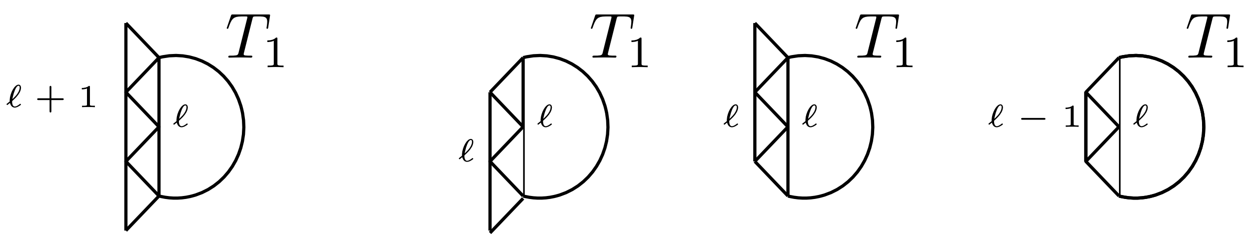
$\longrightarrow$

$$[x]f = [x^0]V = V_0$$

where $V_0 = t(1+V_0)^2$

$$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$$

Parallelogram polyominoes of size $n+1$



Operation u Operation h Operation h' Operation d

where T_1 is a fighting fish with 1 tail

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$f = xV - x^2\frac{V^3}{(1-V)^2}$$

$$V = t(1 + V + \frac{xV^2}{1-V})^2$$

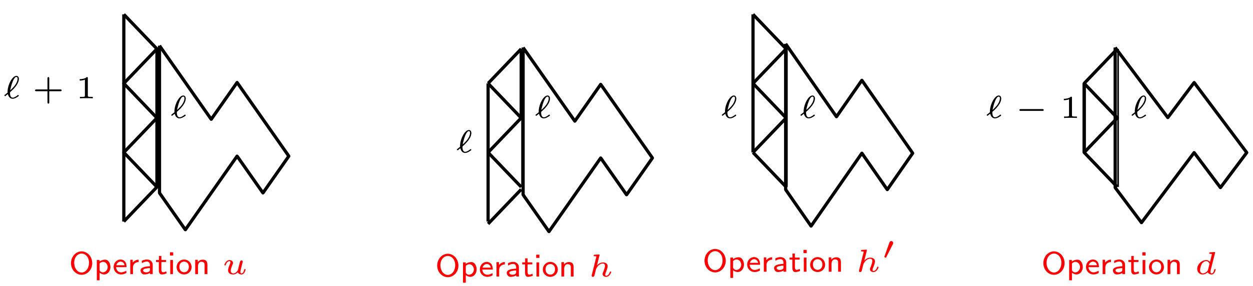
$$\longrightarrow$$

$$[x]f = [x^0]V = V_0$$

where $V_0 = t(1+V_0)^2$

$$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$$

Parallelogram polyominoes of size $n+1$



Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$\begin{aligned} f &= xV - x^2\frac{V^3}{(1-V)^2} \\ V &= t(1+V+\frac{xV^2}{1-V})^2 \end{aligned} \longrightarrow$$

$$[x]f = [x^0]V = V_0$$

where $V_0 = t(1+V_0)^2$

$$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$$

Parallelogram polyominoes of size $n+1$

More generally generating function for fighting fish with size $n+1$ and c tails is rational in the Catalan generating function.

However explicit expressions are not particularly simple

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$\begin{aligned} f &= xV - x^2\frac{V^3}{(1-V)^2} \\ V &= t(1+V+\frac{xV^2}{1-V})^2 \end{aligned} \longrightarrow$$

$[x]f = [x^0]V = V_0$

where $V_0 = t(1+V_0)^2$

$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$

Parallelogram polyominoes of size $n+1$

More generally generating function for fighting fish with size $n+1$ and c tails is rational in the Catalan generating function.

However explicit expressions are not particularly simple

The number of fighting fish with size $n+1$ and a marked tail is $\frac{1}{n}\binom{3n-2}{n-1}$

$\frac{df}{dx}|_{x=1}$ + Lagrange inversion formula

Enumeration wrt semi-perimeter and number of tails.

The same approach can be applied to (re)derive the number of fighting fish of size $n+1$ with one tail.

$$\begin{aligned} f &= xV - x^2\frac{V^3}{(1-V)^2} \\ V &= t(1+V+\frac{xV^2}{1-V})^2 \end{aligned} \longrightarrow$$

$$[x]f = [x^0]V = V_0$$

where $V_0 = t(1+V_0)^2$

$$[xt^n]f = \frac{1}{n+1}\binom{2n}{n}$$

Parallelogram polyominoes of size $n+1$

More generally generating function for fighting fish with size $n+1$ and c tails is rational in the Catalan generating function.

However explicit expressions are not particularly simple

The number of fighting fish with size $n+1$ and a marked tail is $\frac{1}{n}\binom{3n-2}{n-1}$

$\frac{df}{dx}|_{x=1}$ + Lagrange inversion formula

The average number of tails of fighting fish of size $n+1$ is

$$\frac{[x^n]\frac{df}{dx}|_{x=1}}{[x^n]f} = \frac{(n+1)(2n+1)}{3(3n-1)}$$

Enumeration of fighting fish wrt the area

We are going to count fighting fish weighted by their area.

Let $A \equiv A(x,t)$ be the total area generating function.

Then $A(x,t) = \sum_F a(F)t^{n(F)} = \frac{\partial(qf(q))}{\partial q}|_{q=1} = f + \frac{\partial(f(q))}{\partial q}|_{q=1}$

Enumeration of fighting fish wrt the area

We are going to count fighting fish weighted by their area.

Let $A \equiv A(x,t)$ be the total area generating function.

Then $A(x,t) = \sum_F a(F) t^{n(F)} = \frac{\partial(qf(q))}{\partial q}|_{q=1} = f + \frac{\partial(f(q))}{\partial q}|_{q=1}$

Let us consider the master equation

$$T(v,q) = \quad tvq(T(vq^2,q)+x)(T(v,q)+1) + 2tT(vq^2,q)(T(v,q)+1)$$

$$+\frac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

Enumeration of fighting fish wrt the area

We are going to count fighting fish weighted by their area.

Let $A \equiv A(x,t)$ be the total area generating function.

Then $A(x,t) = \sum_F a(F)t^{n(F)} = \frac{\partial(qf(q))}{\partial q}|_{q=1} = f + \frac{\partial(f(q))}{\partial q}|_{q=1}$

Let us consider the master equation

$$T(v,q) = \quad tvq(T(vq^2,q)+x)(T(v,q)+1) + 2tT(vq^2,q)(T(v,q)+1)$$

$$+\frac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

By deriving with respect to q and by setting $q=1$ we obtain

$$(1 - tv(T(v,1)+x) - 2tT(v,1) - \tfrac{t}{v}(T(v,1)-vf) - (tv+2t+\tfrac{t}{v})(T(v)+1))\frac{\partial T}{\partial q}(v,1)$$

$$= (T(v,1)+1)$$

$$\left((tv+2t+\tfrac{t}{v})\cdot 2v\frac{\partial T}{\partial v}(v,1) + tv(T(v,1)+x) - \tfrac{t}{v}(T(v,1)-vf) - 2tf - t\frac{\partial f}{\partial q}(1)\right)$$

Enumeration of fighting fish wrt the area

We are going to count fighting fish weighted by their area.

Let $A \equiv A(x,t)$ be the total area generating function.

Then $A(x,t) = \sum_F a(F) t^{n(F)} = \frac{\partial(qf(q))}{\partial q}|_{q=1} = f + \frac{\partial(f(q))}{\partial q}|_{q=1}$

Let us consider the master equation

$$T(v,q) = tvq(T(vq^2,q)+x)(T(v,q)+1) + 2tT(vq^2,q)(T(v,q)+1)$$
$$+\frac{t}{vq}(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

By setting $v = V$ we have

$$(tV + 2t + \tfrac{t}{V}) \cdot 2V \frac{\partial T}{\partial v}(V,1) + tV(T(V,1)+x) - \tfrac{t}{V}(T(V,1) - Vf) - 2tf - t\frac{\partial f}{\partial q}(1)$$

To obtain $\frac{\partial T}{\partial v}(V,1)$ we apply again the kernel method.

The area generating function

The generating function $A \equiv A(x,t)$ for the total area of fighting fish with size $n+1$ satisfies

$$-V(1-V)^2A^2 + 2(1-V)^2(1-V^2+xV^2)A - 4xV(1-V^2+xV^2) = 0$$

The area generating function

The generating function $A \equiv A(x,t)$ for the total area of fighting fish with size $n+1$ satisfies

$$-V(1-V)^2A^2 + 2(1-V)^2(1-V^2+xV^2)A - 4xV(1-V^2+xV^2) = 0$$

Extracting the coefficient of x, $[x]A = A_1$, yields

$$2(1-V_0)^2(1-V_0^2)A_1 - 4V_0(1-V_0^2) = 0$$

where $V_0 = [x^0]V$ is a Catalan generating function satisfying $V_0 = t(1+V_0)^2$ we obtain the generating function for the total area of parallelogram polyominoes

$A_1 = \frac{2V_0}{(1-V_0)^2} = \frac{2t}{1-4t}$ The simplification to a rational function of t is a well-known feature of parallelogram polyominoes.

The area generating function

The generating function $A \equiv A(x,t)$ for the total area of fighting fish with size $n+1$ satisfies

$$-V(1-V)^2A^2 + 2(1-V)^2(1-V^2+xV^2)A - 4xV(1-V^2+xV^2) = 0$$

Extracting the coefficient of x, $[x]A = A_1$, yields

$$2(1-V_0)^2(1-V_0^2)A_1 - 4V_0(1-V_0^2) = 0$$

where $V_0 = [x^0]V$ is a Catalan generating function satisfying $V_0 = t(1+V_0)^2$
we obtain the generating function for the total area of parallelogram polyominoes

$$A_1 = \frac{2V_0}{(1-V_0)^2} = \frac{2t}{1-4t}$$

The simplification to a rational function of t is a well-known feature of parallelogram polyominoes.

The average area for parallelogram polyominoes of size n is $4^n/C_n$
where C_n are the Catalan numbers.

The average area for parallelogram polyominoes of size n scales like $n^{\frac{3}{2}}$

The area generating function

The generating function $A \equiv A(x,t)$ for the total area of fighting fish with size $n+1$ satisfies

$$-V(1-V)^2A^2 + 2(1-V)^2(1-V^2+xV^2)A - 4xV(1-V^2+xV^2) = 0$$

Extracting the coefficient of x, $[x]A = A_1$, yields

$$2(1-V_0)^2(1-V_0^2)A_1 - 4V_0(1-V_0^2) = 0$$

where $V_0 = [x^0]V$ is a Catalan generating function satisfying $V_0 = t(1+V_0)^2$ we obtain the generating function for the total area of parallelogram polyominoes

$$A_1 = \frac{2V_0}{(1-V_0)^2} = \frac{2t}{1-4t}$$

The simplification to a rational function of t is a well-known feature of parallelogram polyominoes.

The average area for parallelogram polyominoes of size n is $4^n/C_n$ where C_n are the Catalan numbers.

The average area for parallelogram polyominoes of size n scales like $n^{\frac{3}{2}}$

The generating function of the total area of fighting fish with c tails and size $n+1$ is a rational function of the Catalan generating function V_0

The area generatin function

The average area A_n of fighting fish with size $n + 1$ grows like $n^{\frac{5}{4}}$

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

Indeed, let us reconsider the equation for V, f, and A,

$$V = \frac{t}{(1-V)^2}$$

and get asymptotics by singularity analysis

$$f = V - \frac{V^3}{(1-V)^2}$$

$$A = \frac{1}{V} - \sqrt{\frac{(1+V)(1-3V)}{V(1-V)}}$$

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

Indeed, let us reconsider the equation for V, f, and A, and get asymptotics by singularity analysis

$$\begin{aligned} V &= \tfrac{t}{(1-V)^2} \\ f &= V - \tfrac{V^3}{(1-V)^2} \\ A &= \tfrac{1}{V} - \sqrt{\tfrac{(1+V)(1-3V)}{V(1-V)}} \end{aligned}$$

We obtain:

$$[t^n]A \underset{n\to\infty}{\sim} cte \cdot n^{-\frac{5}{4}} t_c^{-n}$$

$$[t^n]f \underset{n\to\infty}{\sim} cte \cdot n^{-\frac{5}{2}} t_c^{-n}$$

Then the average area is

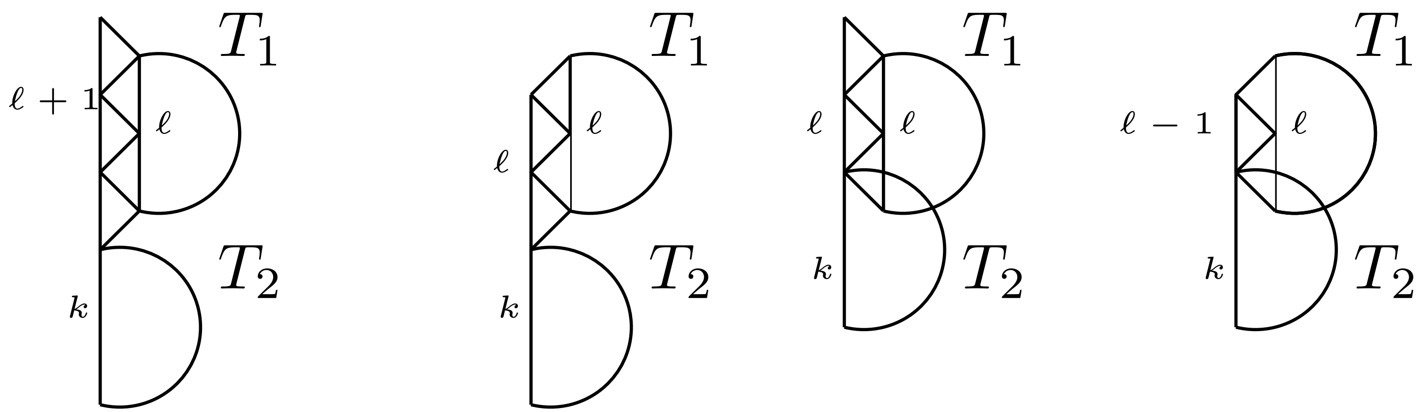
$$\frac{[t^n]A}{[t^n]f} \underset{n\to\infty}{\sim} cte \cdot n^{\frac{5}{4}}$$

A refinement of the main formula

$T(v,q,x,t) = \sum_{T \in \mathcal{FT}} a^{l(T)} b^{r(T)} v^{h(T)} q^{a(T)} x^{c(T)} t^{n(T)}$ denote the generating function of fish tails, where

$l(T)$ is the number of upper-left free edges of T

$r(T)$ is the number of upper-right edges of T



Operation u | Operation h | Operation h' | Operation d

$$T(v,q) = \quad tvqb(T(vq^2,q)+x)(T(v,q)+1) + aT(vq^2,q)(T(v,q)+1) +$$

$$+bT(vq^2,q)(T(v,q)+1) + \frac{t}{vq}a(T(vq^2,q) - vq^2 f(q))(T(v,q)+1)$$

$$\#\left\{ \begin{array}{c} \text{fighting fish with} \\ i \text{ top left and } j \text{ top right edges} \end{array} \right\} = \frac{1}{(2i+j-1)(2j+i-1)}\binom{2i+j-1}{i}\binom{2j+i-1}{j}$$

An algebraic decomposition for parallelogram polyominoes.

Since the gf function f of fighting fish is algebraic we would like to find an algebraic decomposition.

An algebraic decomposition for parallelogram polyominoes.

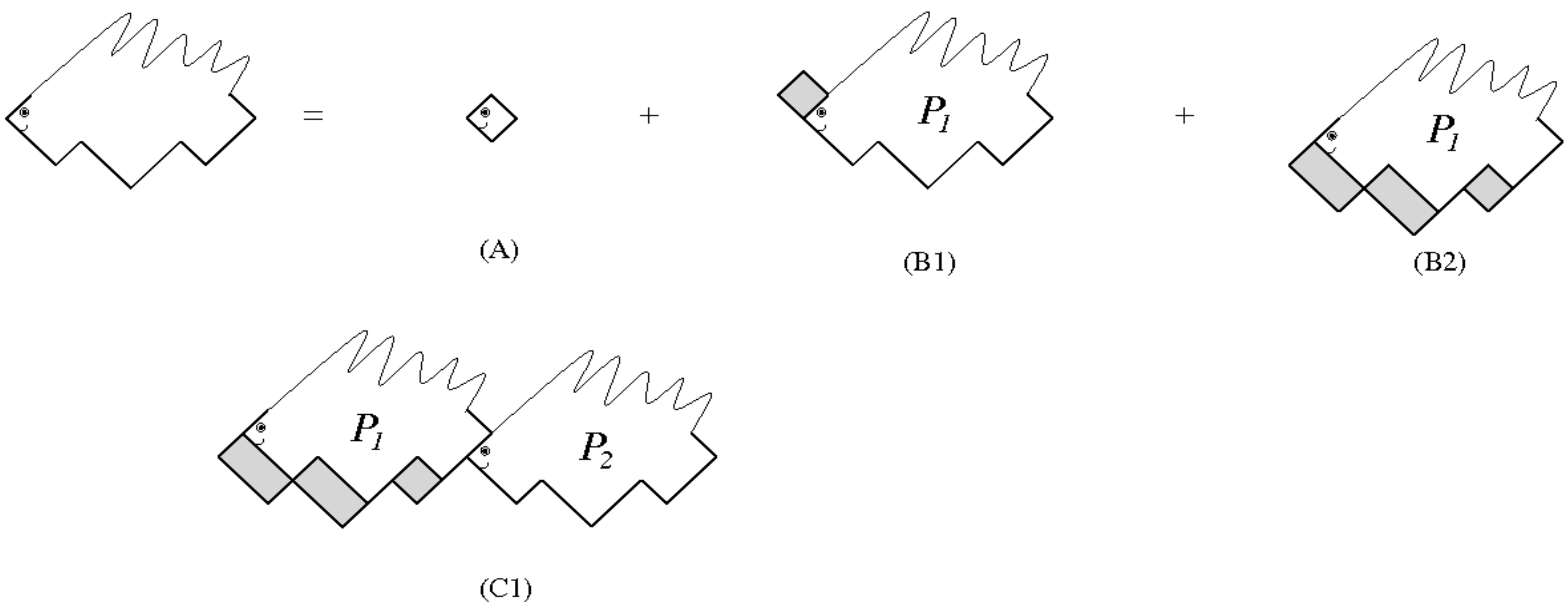
Since the gf function f of fighting fish is algebraic we would like to find an algebraic decomposition.

We try a new decomposition starting from
a grammar-like decomposition on parallelogram polyominoes.

An algebraic decomposition for parallelogram polyominoes.

Since the gf function f of fighting fish is algebraic we would like to find an algebraic decomposition.

We try a new decomposition starting from
a grammar-like decomposition on parallelogram polyominoes.

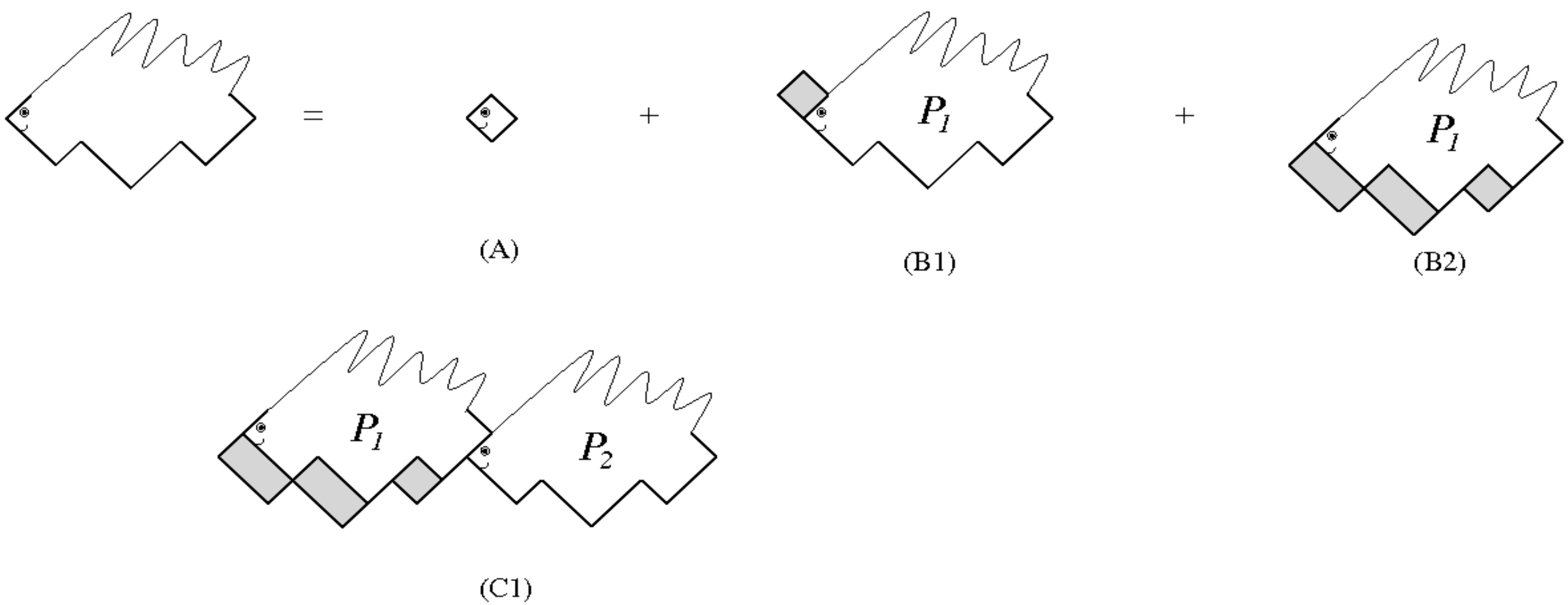


The wasp-waist decomposition for parallelogram polyominoes

An algebraic decomposition for parallelogram polyominoes.

Since the gf function f of fighting fish is algebraic we would like to find an algebraic decomposition.

We try a new decomposition starting from
a grammar-like decomposition on parallelogram polyominoes.



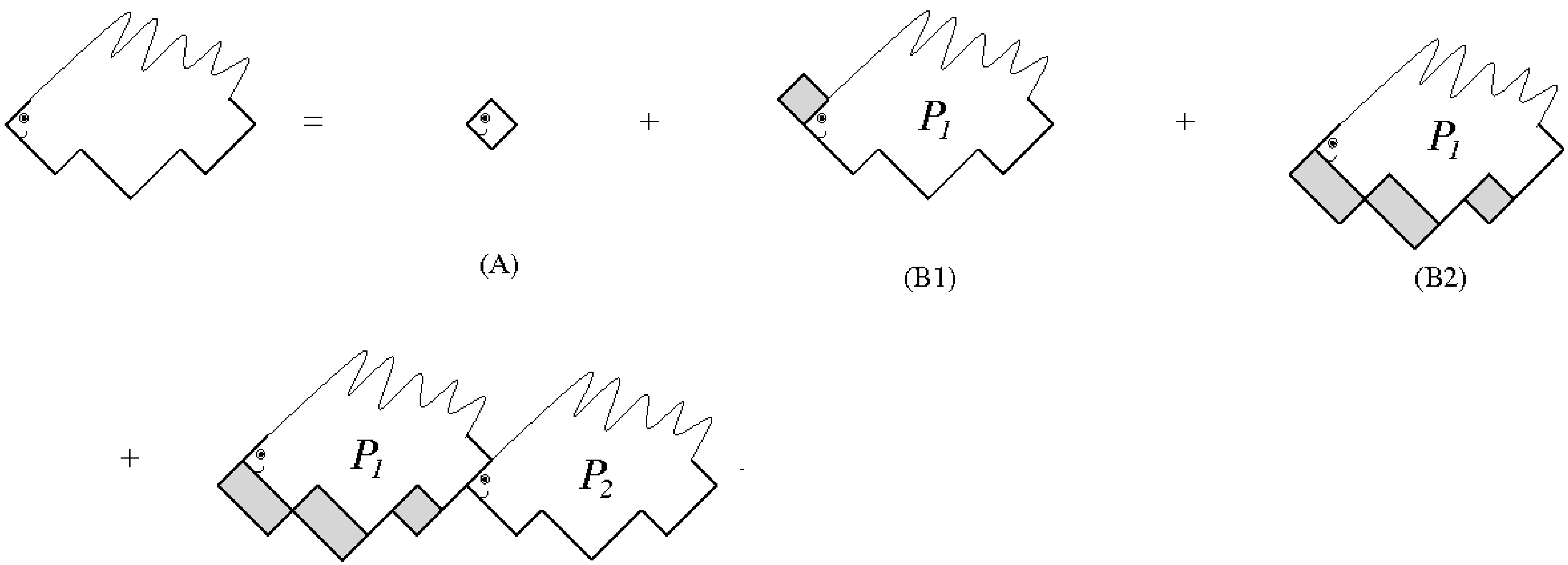
The wasp-waist decomposition for parallelogram polyominoes

Let $P = \sum_P t^{|P|}$ be the GF of parallelogram polyominoes according to the size, then $\quad P = t + 2tP + tP^2$

A new decomposition

Extend the *wasp-waist decomposition* of parallelogram polyominoes:

- remove one cell at the bottom of each diagonal, from left to right along the fin, until this creates a cut

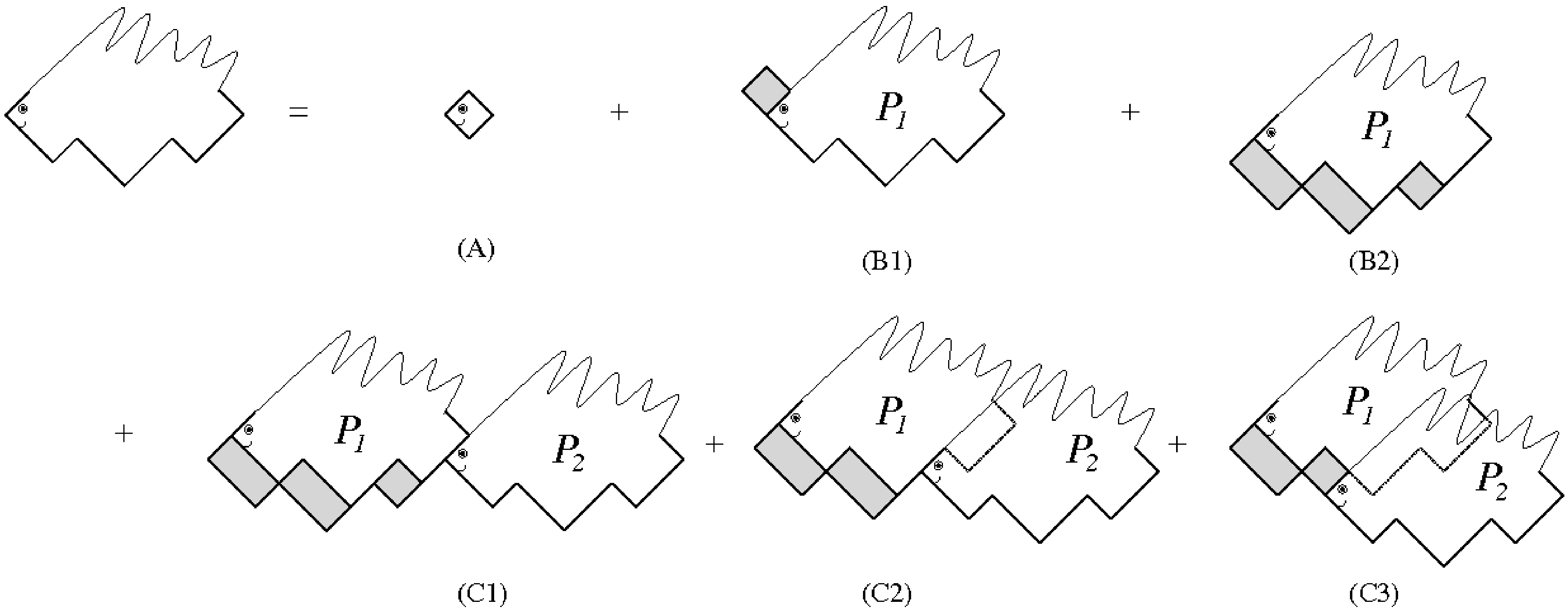


(A) (B1) (B2) (C1)

A new decomposition

Extend the *wasp-waist decomposition* of parallelogram polyominoes:

remove one cell at the bottom of each diagonal, from left to right along the fin, until this creates a cut



Two more cases must be considered for fighting fish...

A glipse of the proof

Let $F(u) = \sum_f t^{|f|} u^{\text{fin}(f)} x^{\text{tail}(f)-1}$ be the GF of fighting fish according to the size, fin length and number of extra tails.
Then

$$F(u) = tu(1 + F(u))^2 + xtuF(u)\frac{F(1) - F(u)}{1 - u} \quad \text{with } f = F(1).$$

A glipse of the proof

Let $F(u) = \sum_f t^{|f|} u^{\mathrm{fin}(f)} x^{\mathrm{tail}(f)-1}$ be the GF of fighting fish according to the size, fin length and number of extra tails.
Then

$$F(u) = tu(1 + F(u))^2 + xtuF(u)\frac{F(1) - F(u)}{1 - u} \quad \text{with } f = F(1).$$

Case $x = 0$. Fish with one tail, *ie* parallelogram polyominoes:
we have the usual algebraic equation for the GF of Catalan numbers.

A glipse of the proof

Let $F(u) = \sum_f t^{|f|} u^{\text{fin}(f)} x^{\text{tail}(f)-1}$ be the GF of fighting fish according to the size, fin length and number of extra tails.
Then

$$F(u) = tu(1+F(u))^2 + xtuF(u)\frac{F(1)-F(u)}{1-u} \quad \text{with } f = F(1).$$

.

Case $x = 0$. Fish with one tail, *ie* parallelogram polyominoes:
we have the usual algebraic equation for the GF of Catalan numbers.

But in the general case we have again a polynomial equation with one catalytic variable...

$\Rightarrow$ The question to find a direct algebraic decomposition of fighting fish remain.

Bijections and parameter equidistributions?

Sloane's Online Encyclopedia of Integer Sequences

$$\#\left\{\begin{array}{c}\text{fighting fish}\\ \text{with semi-perimeter } n+1\end{array}\right\} = \frac{2}{(n+1)(2n+1)}\binom{3n}{n}$$

1, 2, 6, 91, 408, 1938...

This integer sequence was already in Sloane's OEIS!

The number of fighting fish of size $n+1$ (with i left and j down top edges) is equal to the number of:

- Two-stack sortable permutations of $\{1, \ldots, n\}$ (i ascending and j descending runs) (West, Zeilberger, Bona, 90's)

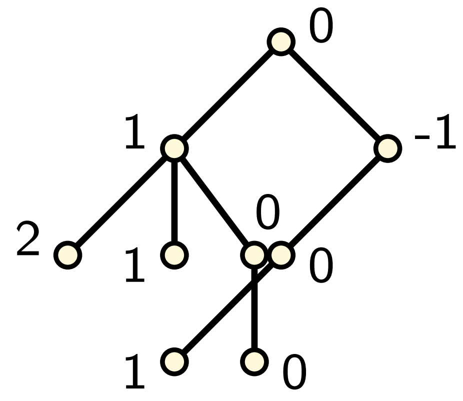
- Rooted non separable planar maps with n edges ($i+1$ vertices, $j+1$ faces) (Tutte, Mullin and Schellenberg, 60's)

- Left ternary trees with n noeuds ($i+1$ even, j odd vertices) (Del Lungo, Del Ristoro, Penaud, 1999)

Left ternary trees and further equidistributions

Natural embedding of a ternary tree:

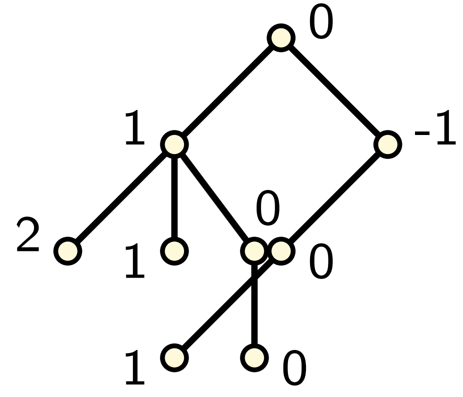
- root vertex has label 0
- vertex with label $i \Rightarrow$ left child $i+1$,
  central child i, right child $i-1$.



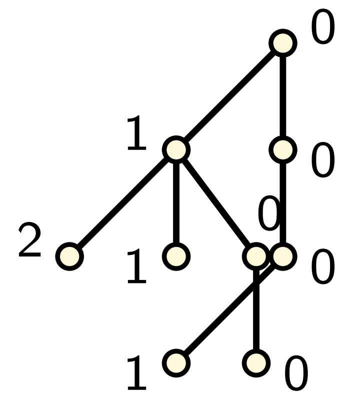
Left ternary trees and further equidistributions

Natural embedding of a ternary tree:

- root vertex has label 0
- vertex with label $i \Rightarrow$ left child $i+1$, central child i, right child $i-1$.



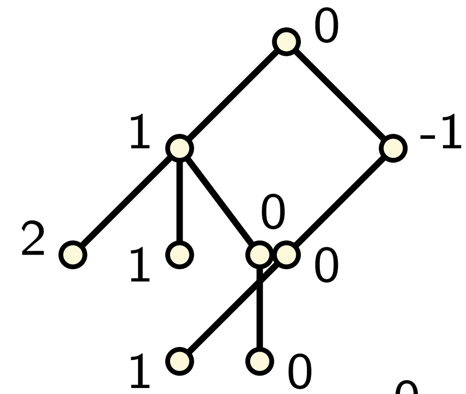
Left ternary tree $=$ ternary tree without negative labels.



Left ternary trees and further equidistributions

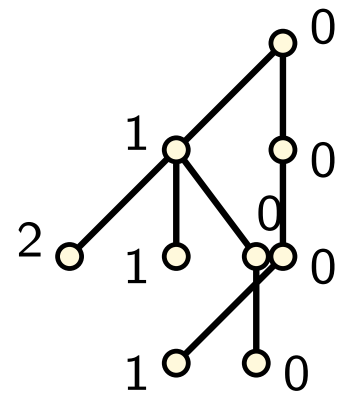
Natural embedding of a ternary tree:

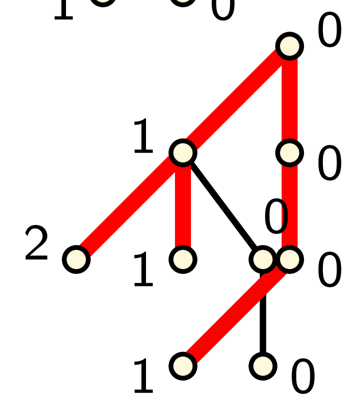
- root vertex has label 0
- vertex with label $i \Rightarrow$ left child $i+1$, central child i, right child $i-1$.



Left ternary tree $=$ ternary tree without negative labels.

Core $=$ binary subtree of the root after pruning all right edges

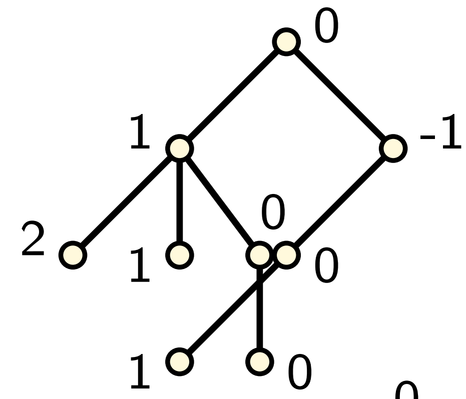




Left ternary trees and further equidistributions

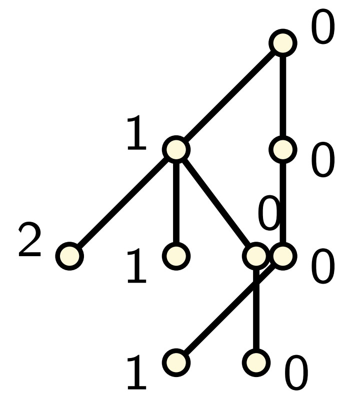
Natural embedding of a ternary tree:

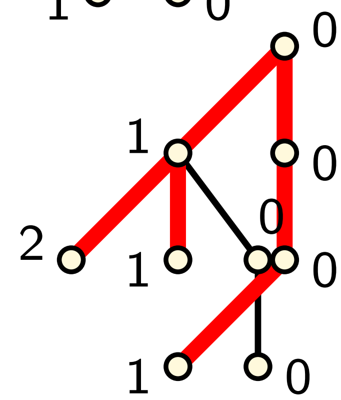
- root vertex has label 0
- vertex with label $i \Rightarrow$ left child $i+1$, central child i, right child $i-1$.



Left ternary tree $=$ ternary tree without negative labels.

Core $=$ binary subtree of the root after pruning all right edges



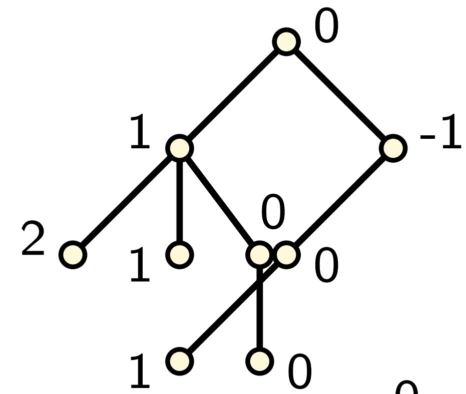


Theorem (DGRS 2016): The number of fighting fish with size $n+1$ and fin length k equals the number of left ternary trees with n nodes and core size k.

Left ternary trees and further equidistributions

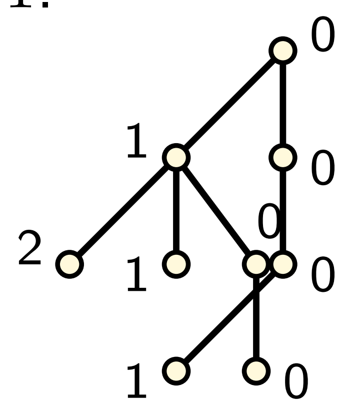
Natural embedding of a ternary tree:

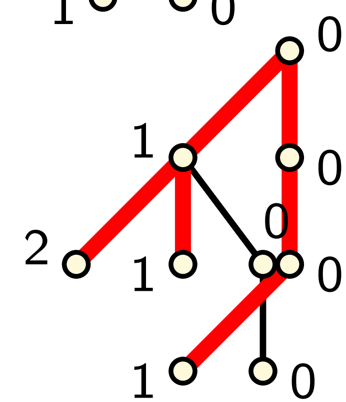
- root vertex has label 0
- vertex with label $i \Rightarrow$ left child $i+1$,
  central child i, right child $i-1$.



Left ternary tree $=$ ternary tree without negative labels.

Core $=$ binary subtree of the root after pruning all right edges

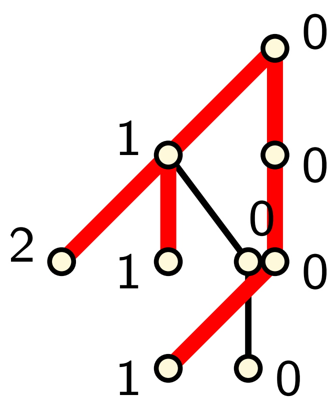


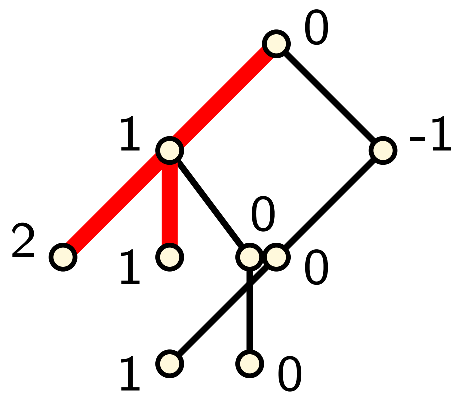


Theorem (DGRS 2016): The number of fighting fish with size $n+1$ and fin length k equals the number of left ternary trees with n nodes and core size k.

Proof? We computed the gf of fighting fish wrt size and fin length. Compute the gf of left ternary trees wrt size and core size...

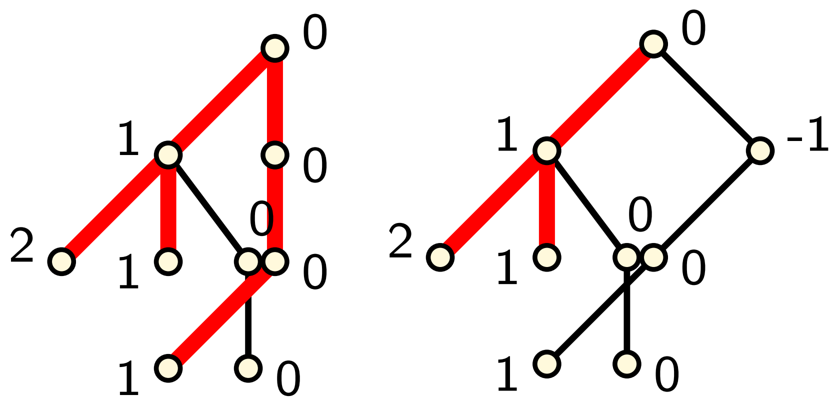
Left ternary trees and further equidistributions





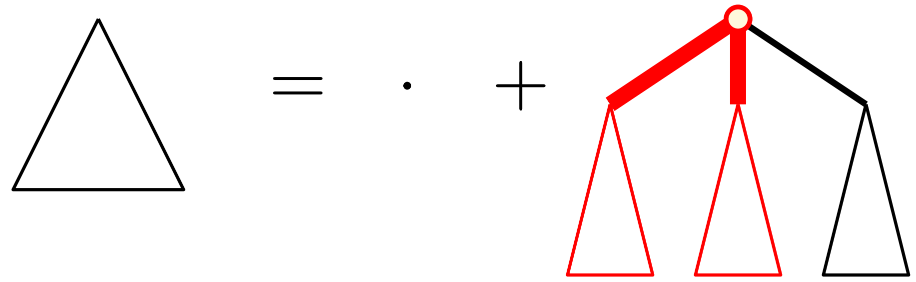
Core = binary subtree of the root after pruning all right edges

Left ternary trees and further equidistributions

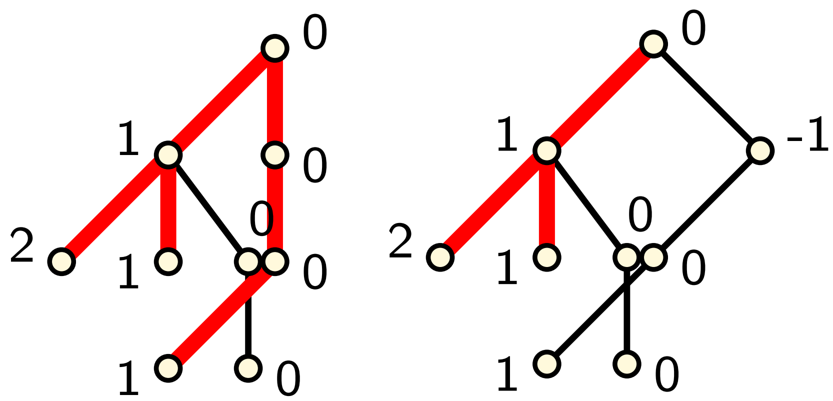


Easy for ternary trees:

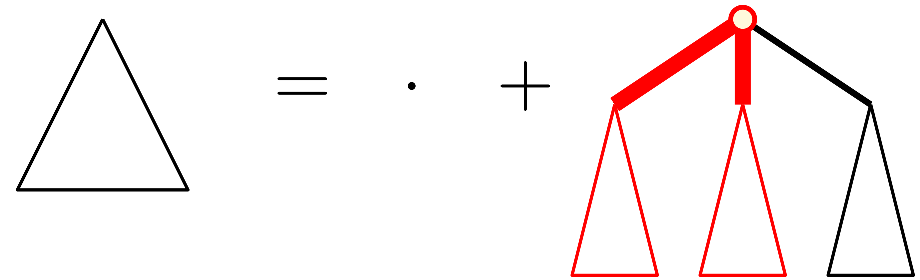
Core = binary subtree of the root after pruning all right edges



Left ternary trees and further equidistributions



Core $=$ binary subtree of the root after pruning all right edges

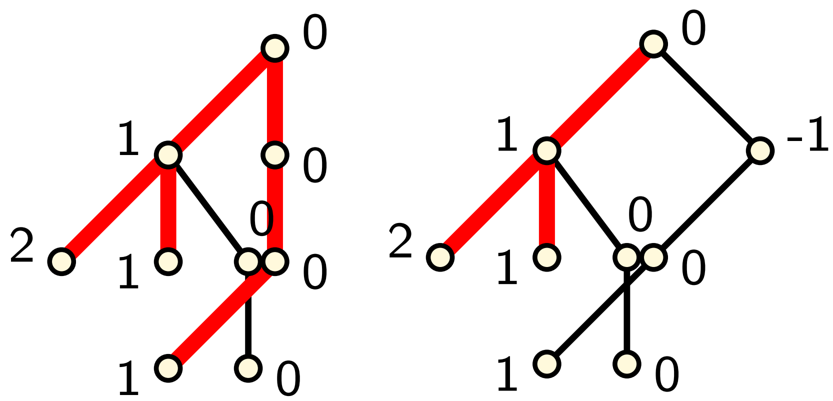


Easy for ternary trees:

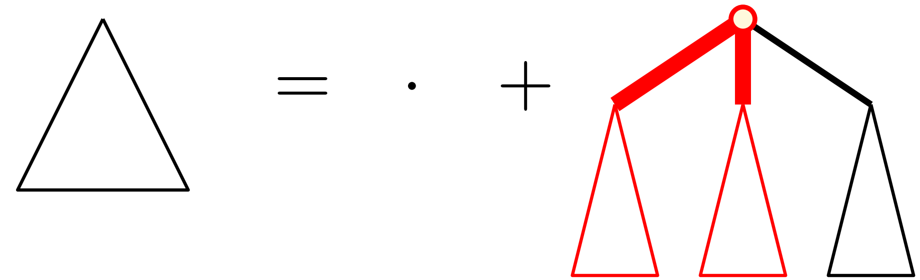
Proposition The GF $\tau(u)$ of ternary trees wrt size and core size is

$$\tau(u) = 1 + tu\tau(u)^2\tau(1) \quad \text{generalizing} \quad \tau(1) = 1 + t\tau(1)^3$$

Left ternary trees and further equidistributions



Core $=$ binary subtree of the root
after pruning all right edges



Easy for ternary trees:

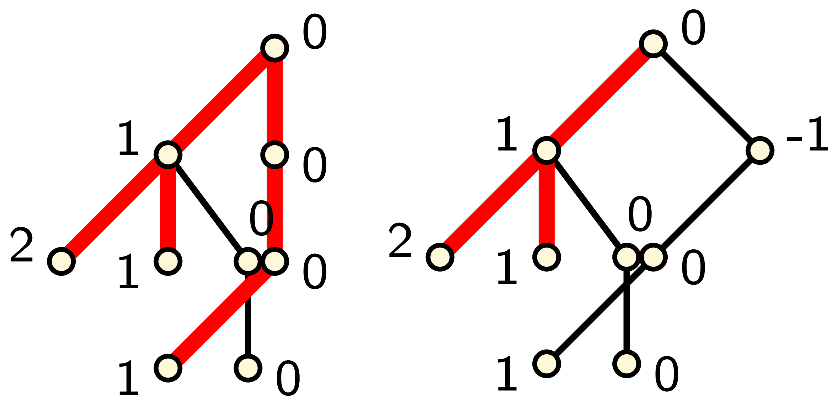
Proposition The GF $\tau(u)$ of ternary trees wrt size and core size is

$$\tau(u) = 1 + tu\tau(u)^2\tau(1) \quad \text{generalizing} \quad \tau(1) = 1 + t\tau(1)^3$$

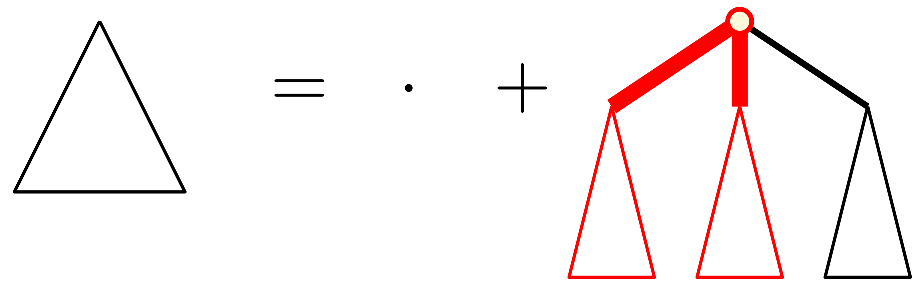
But: No known simple decomposition of left ternary trees.

Known decompositions are complex and do not preserve core.

Left ternary trees and further equidistributions



Core $=$ binary subtree of the root after pruning all right edges



Easy for ternary trees:

Proposition The GF $\tau(u)$ of ternary trees wrt size and core size is

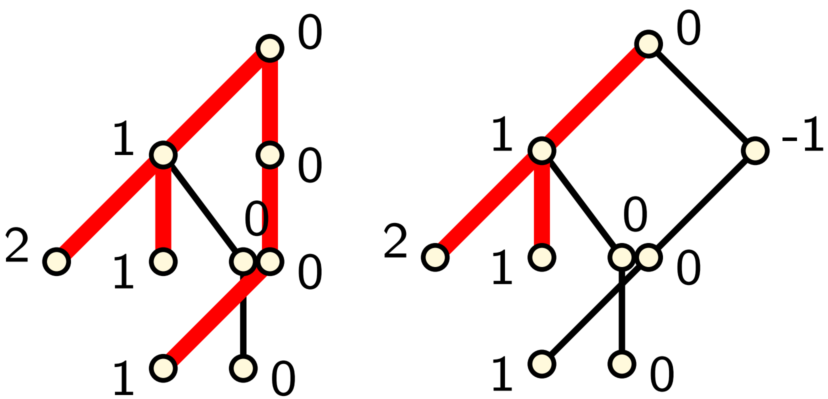
$$\tau(u) = 1 + tu\tau(u)^2\tau(1) \quad \text{generalizing} \quad \tau(1) = 1 + t\tau(1)^3$$

But: No known simple decomposition of left ternary trees.

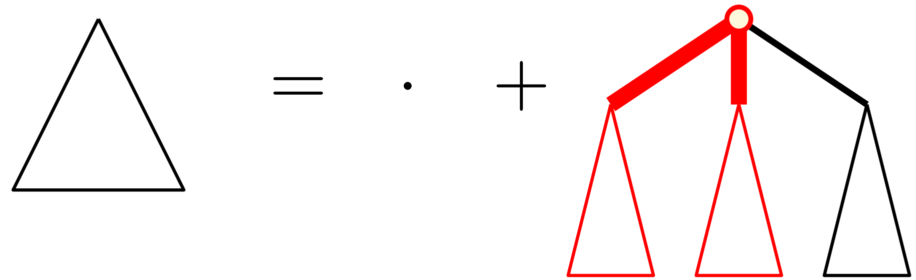
Known decompositions are complex and do not preserve core.

Idea: Solve a more general problem...

Left ternary trees and further equidistributions



Core = binary subtree of the root after pruning all right edges



Easy for ternary trees:

Proposition The GF $\tau(u)$ of ternary trees wrt size and core size is

$$\tau(u) = 1 + tu\tau(u)^2\tau(1) \quad \text{generalizing} \quad \tau(1) = 1 + t\tau(1)^3$$

But: No known simple decomposition of left ternary trees.

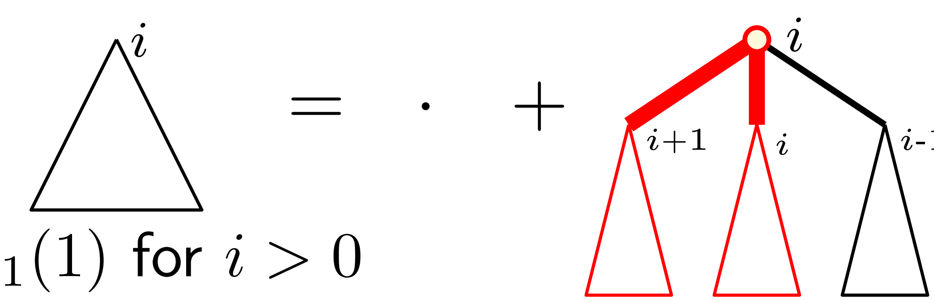
Known decompositions are complex and do not preserve core.

Idea: Solve a more general problem...

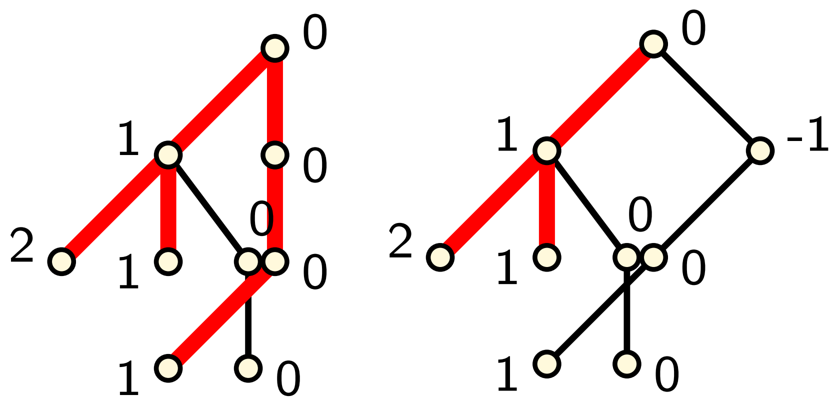
Proposition The GFs wrt size and core size of left ternary trees with root label i satisfy

$$\tau_0(u) = 1 + tu\tau_1(u)\tau_0(u)$$

$$\tau_i(u) = 1 + tu\tau_{i+1}(u)\tau_i(u)\tau_{i-1}(1) \text{ for } i > 0$$



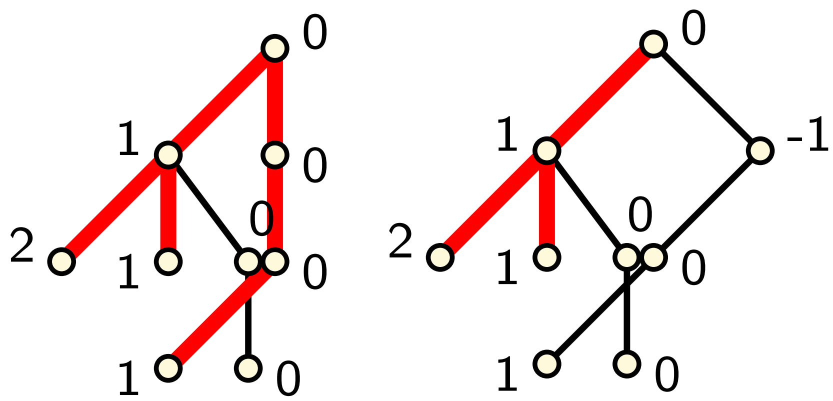
Left ternary trees and further equidistributions



Core $=$ binary subtree of the root after pruning all right edges

The solution of the previous infinite system of equation is known for $u = 1$:

Left ternary trees and further equidistributions



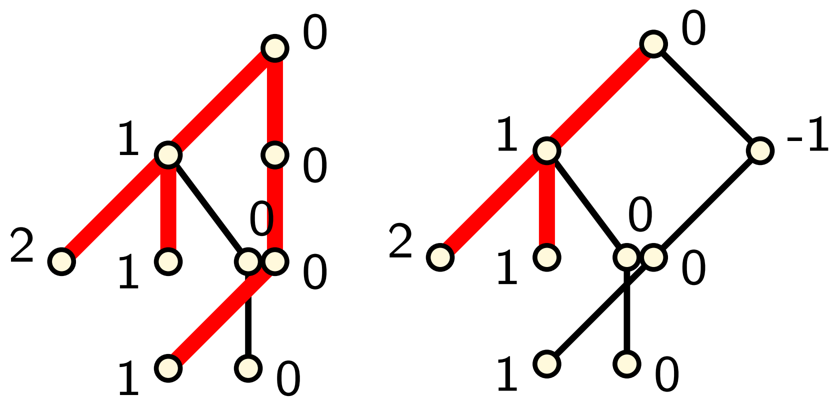
Core $=$ binary subtree of the root after pruning all right edges

The solution of the previous infinite system of equation is known for $u = 1$:

Theorem (Di Francesco 05, Kuba 11) The size GF of left ternary trees with root label i is

$$\tau_i = \tau \frac{(1-X^{i+5})}{(1-X^{i+4})} \frac{(1-X^{i+2})}{(1-X^{i+3})} \quad \text{where} \left\{ \begin{array}{ccc} \tau & = & 1 + t\tau^3 \\ X & = & (1 + X + X^2)\frac{\tau-1}{\tau} \end{array} \right. .$$

Left ternary trees and further equidistributions



Core = binary subtree of the root after pruning all right edges

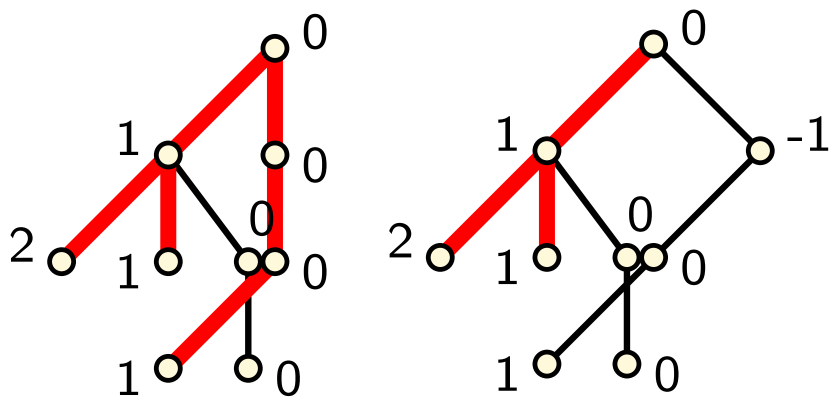
The solution of the previous infinite system of equation is known for $u = 1$:

Theorem (Di Francesco 05, Kuba 11) The size GF of left ternary trees with root label i is

$$\tau_i = \tau \frac{(1-X^{i+5})}{(1-X^{i+4})} \frac{(1-X^{i+2})}{(1-X^{i+3})} \quad \text{where} \begin{cases} \tau &= 1 + t\tau^3 \\ X &= (1 + X + X^2)\frac{\tau - 1}{\tau} \end{cases}.$$

Proof by guessing the formula and checking it satisfies the recurrence.

Left ternary trees and further equidistributions



Core $=$ binary subtree of the root after pruning all right edges

The solution of the previous infinite system of equation is known for $u = 1$:

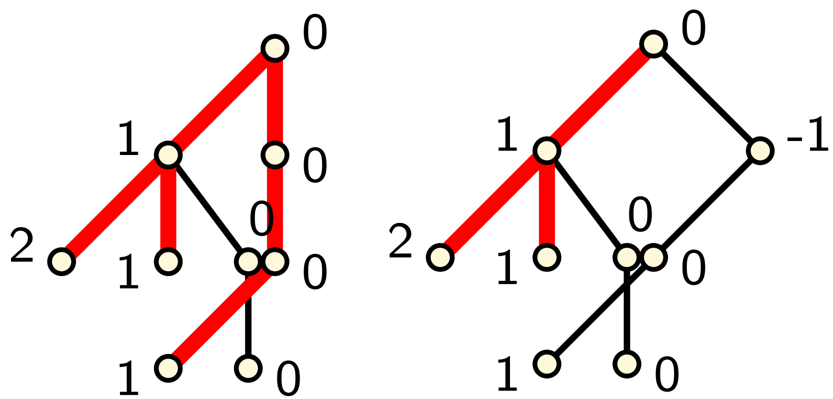
Theorem (Di Francesco 05, Kuba 11) The size GF of left ternary trees with root label i is

$$\tau_i = \tau \frac{(1-X^{i+5})}{(1-X^{i+4})} \frac{(1-X^{i+2})}{(1-X^{i+3})} \quad \text{where} \left\{ \begin{array}{rcl} \tau & = & 1 + t\tau^3 \\ X & = & (1 + X + X^2)\frac{\tau - 1}{\tau} \end{array} \right. .$$

Proof by guessing the formula and checking it satisfies the recurrence.

Case $i = 0$ of this thm gives formula for left ternary trees of size n

Left ternary trees and further equidistributions



Core $=$ binary subtree of the root after pruning all right edges

The solution of the previous infinite system of equation is known for $u = 1$:

Theorem (Di Francesco 05, Kuba 11) The size GF of left ternary trees with root label i is

$$\tau_i = \tau \frac{(1-X^{i+5})}{(1-X^{i+4})} \frac{(1-X^{i+2})}{(1-X^{i+3})} \quad \text{where} \left\{ \begin{array}{lcl} \tau & = & 1 + t\tau^3 \\ X & = & (1 + X + X^2)\frac{\tau - 1}{\tau} \end{array} \right. .$$

Proof by guessing the formula and checking it satisfies the recurrence.

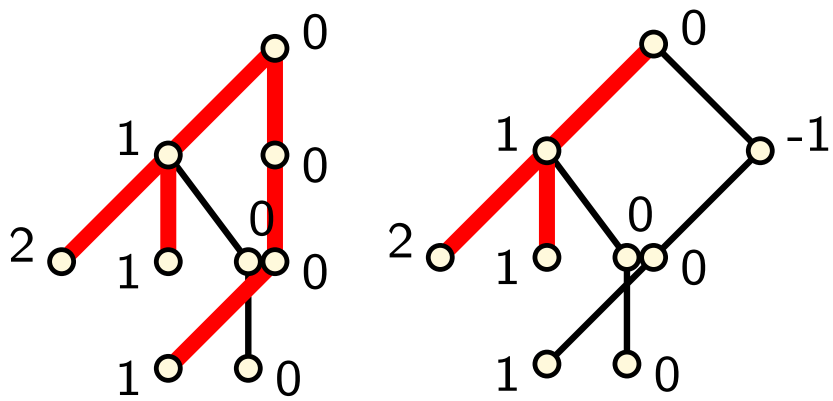
Case $i = 0$ of this thm gives formula for left ternary trees of size n

We extend the theorem and its proof by guessing the bivariate formula

Theorem (DGRS16) The bivariate size and core size GF of left ternary trees with label i is

$$\tau_i(u) = \tau(u) \frac{H_i(u)}{H_{i-1}(u)} \frac{1-X^{i+2}}{1-X^{i+3}} \quad \text{where} \left\{ \begin{array}{rl} \tau(u) = & 1 + tu\tau\tau(u)^2 \\ H_j(u) = & (1 - X^{j+1})XT(u) \\ & -(1 + X)(1 - X^{j+2}) \end{array} \right. .$$

Left ternary trees and further equidistributions



Core $=$ binary subtree of the root after pruning all right edges

The solution of the previous infinite system of equation is known for $u = 1$:

Theorem (DGRS 2016): The number of fighting fish with size $n+1$ and fin length k equals the number of left ternary trees with n nodes and core size k.

Conjecture (DGRS 2016): The previous computation can be refined to prove joined equidistribution of:

fin length $\leftrightarrow$ core size
number of tails $\leftrightarrow$ number of right branches
number of left/right free edges $\leftrightarrow$ number of even/odd labels

Bijections ?

fighting fish

2SS-permutations

left ternary trees

ns planar maps

Bijections ?

fighting fish

2SS-permutations

left ternary trees

ns planar maps

Tutte

recursive decomposition + GF

Bijections ?

fighting fish

2SS-permutations

left ternary trees

Zeilberger

recursive decomposition + GF

ns planar maps

Tutte

recursive decomposition + GF

Bijections ?

fighting fish

2SS-permutations

left ternary trees

Zeilberger

recursive decomposition + GF

.

Goulden-West

(isomorphic recursive decompositions)

ns planar maps

Tutte

recursive decomposition + GF

Bijections ?

fighting fish

2SS-permutations

left ternary trees

Zeilberger

recursive decomposition + GF

.

Del Lungo *et al*

(isomorphic recursive decompositions)

Goulden-West

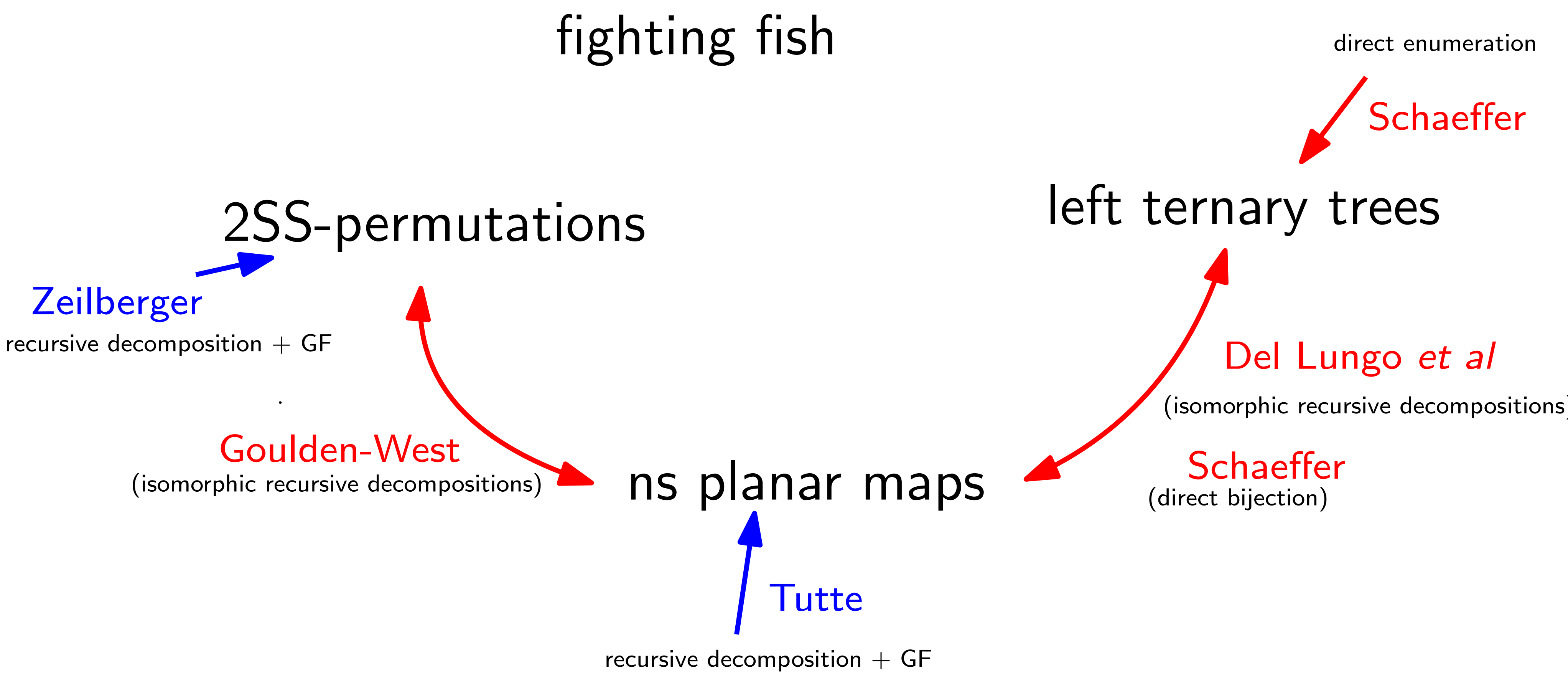
(isomorphic recursive decompositions)

ns planar maps

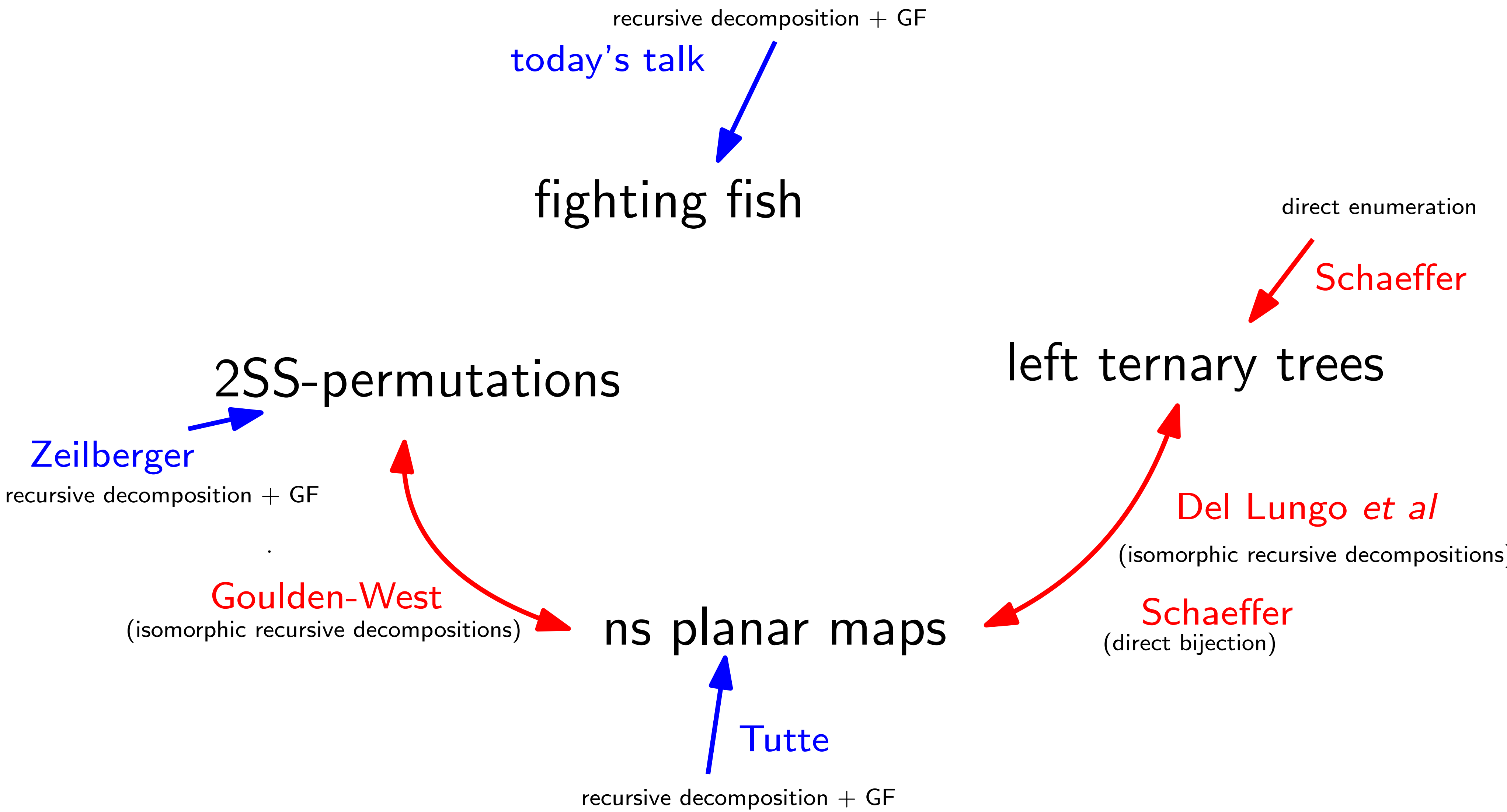
Tutte

recursive decomposition + GF

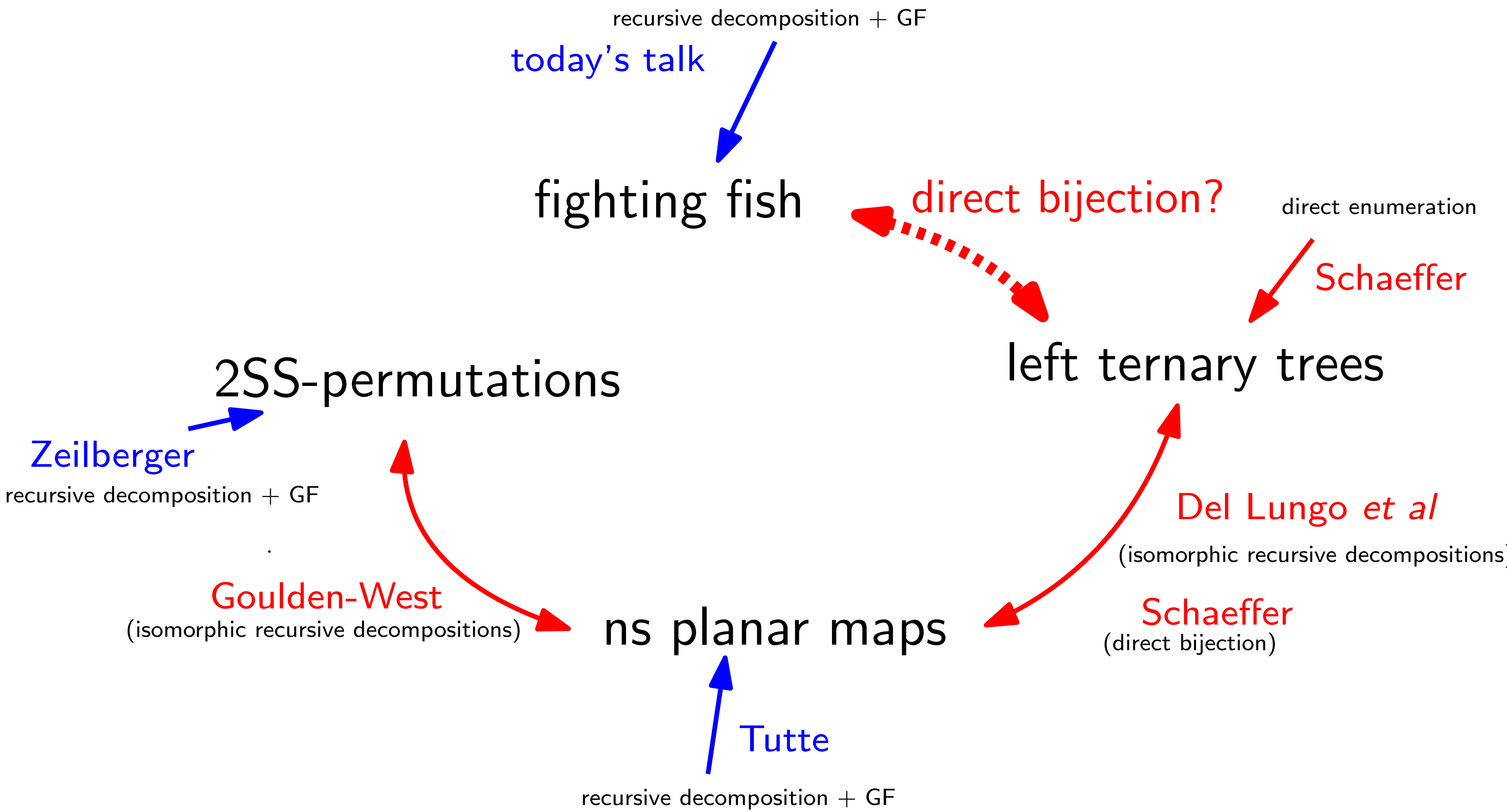
Bijections ?


fighting fish
direct enumeration
Schaeffer
left ternary trees
2SS-permutations
Zeilberger
recursive decomposition + GF
Del Lungo *et al*
(isomorphic recursive decompositions)
Goulden-West
(isomorphic recursive decompositions)
ns planar maps
Schaeffer
(direct bijection)
Tutte
recursive decomposition + GF

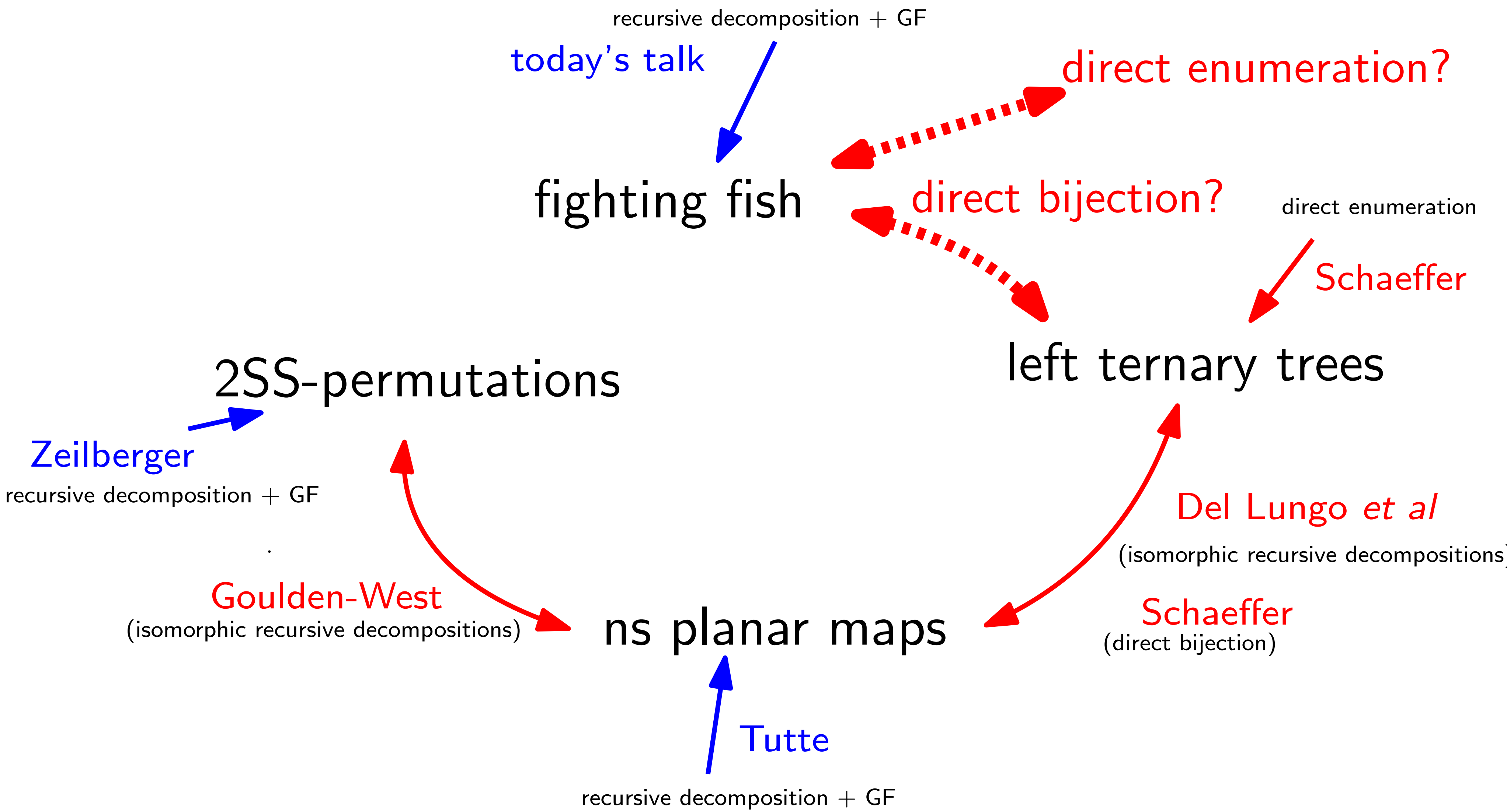
Bijections ?


recursive decomposition + GF
today's talk
fighting fish
direct enumeration
Schaeffer
left ternary trees
2SS-permutations
Zeilberger
recursive decomposition + GF
Del Lungo *et al*
(isomorphic recursive decompositions)
Goulden-West
(isomorphic recursive decompositions)
ns planar maps
Schaeffer
(direct bijection)
Tutte
recursive decomposition + GF

Bijections ?

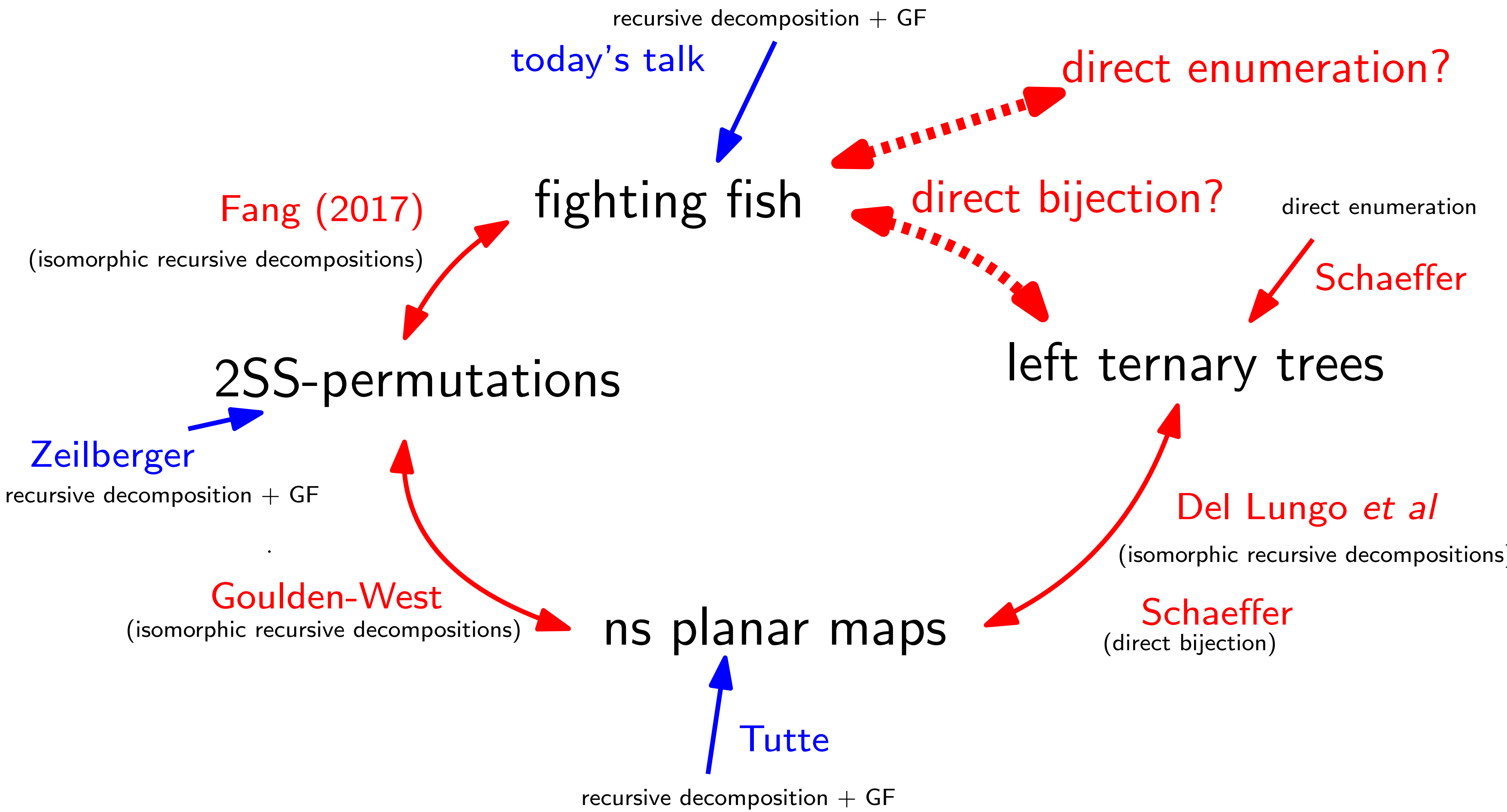

recursive decomposition + GF
today's talk
fighting fish
direct bijection?
direct enumeration
Schaeffer
left ternary trees
2SS-permutations
Zeilberger
recursive decomposition + GF
Del Lungo *et al*
(isomorphic recursive decompositions)
Goulden-West
(isomorphic recursive decompositions)
ns planar maps
Schaeffer
(direct bijection)
Tutte
recursive decomposition + GF

Bijections ?

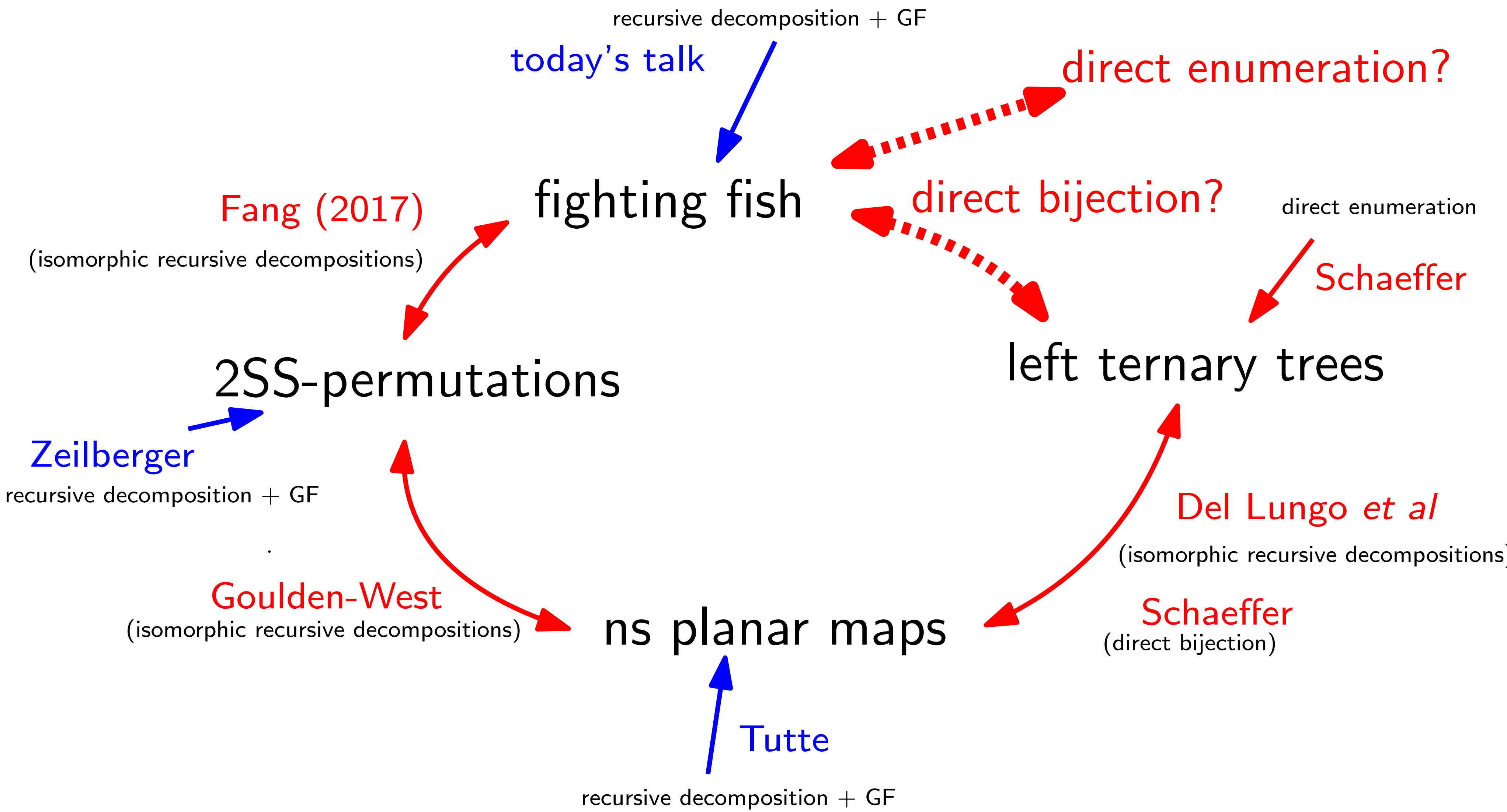


recursive decomposition + GF
today's talk
direct enumeration?
fighting fish
direct bijection?
direct enumeration
Schaeffer
left ternary trees
2SS-permutations
Zeilberger
recursive decomposition + GF
Del Lungo *et al*
(isomorphic recursive decompositions)
Goulden-West
(isomorphic recursive decompositions)
ns planar maps
Schaeffer
(direct bijection)
Tutte
recursive decomposition + GF

Bijections ?


recursive decomposition + GF
today's talk
direct enumeration?
fighting fish
direct bijection?
direct enumeration
Fang (2017)
(isomorphic recursive decompositions)
Schaeffer
2SS-permutations
left ternary trees
Zeilberger
recursive decomposition + GF
Del Lungo *et al*
(isomorphic recursive decompositions)
Goulden-West
(isomorphic recursive decompositions)
ns planar maps
Schaeffer
(direct bijection)
Tutte
recursive decomposition + GF

Bijections ?


recursive decomposition + GF
today's talk
direct enumeration?
fighting fish
direct bijection?
direct enumeration
Fang (2017)
(isomorphic recursive decompositions)
Schaeffer
left ternary trees
2SS-permutations
Zeilberger
recursive decomposition + GF
Del Lungo *et al*
(isomorphic recursive decompositions)
Goulden-West
(isomorphic recursive decompositions)
Schaeffer
(direct bijection)
ns planar maps
Tutte
recursive decomposition + GF

THANK YOU

THANK YOU

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

Indeed, let us reconsider the equation for V, f, and A,

$$V = \tfrac{t}{(1-V)^2}$$

and get asymptotics by singularity analysis

$$f = V - \tfrac{V^3}{(1-V)^2}$$

$$A = \tfrac{1}{V} - \sqrt{\tfrac{(1+V)(1-3V)}{V(1-V)}}$$

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

Indeed, let us reconsider the equation for V, f, and A,

$$\begin{aligned} V &= \frac{t}{(1-V)^2} \\ f &= V - \frac{V^3}{(1-V)^2} \\ A &= \frac{1}{V} - \sqrt{\frac{(1+V)(1-3V)}{V(1-V)}} \end{aligned}$$

and get asymptotics by singularity analysis

$V =$ generating function of a simple family of trees.

Square root singularity expansion near dominant singularity

$$V = \frac{1}{3} - cte\sqrt{(1-\frac{t}{t_c})} + O(1-\frac{t}{t_c})$$

with $t_c = \frac{4}{27}$

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

Indeed, let us reconsider the equation for V, f, and A,

$$\begin{aligned} V &= \tfrac{t}{(1-V)^2} \\ f &= V - \tfrac{V^3}{(1-V)^2} \\ A &= \tfrac{1}{V} - \sqrt{\tfrac{(1+V)(1-3V)}{V(1-V)}} \end{aligned}$$

and get asymptotics by singularity analysis

$V =$ generating function of a simple family of trees.

Square root singularity expansion near dominant singularity

$$V = \frac{1}{3} - cte\sqrt{(1-\frac{t}{t_c})} + O(1-\frac{t}{t_c})$$

with $t_c = \frac{4}{27}$

$$f = \frac{1}{4} - \frac{3}{4}(1-\frac{t}{t_c}) + \frac{\sqrt{3}}{2}(1-\frac{t}{t_c})^{\frac{3}{2}} + O((1-\frac{t}{t_c})^2)$$

The area generatin function

The average area A_n of fighting fish with size $n+1$ grows like $n^{\frac{5}{4}}$

Indeed, let us reconsider the equation for V, f, and A,

$$V = \frac{t}{(1-V)^2}$$

and get asymptotics by singularity analysis

$$f = V - \frac{V^3}{(1-V)^2}$$

$V =$ generating function of a simple family of trees.

$$A = \frac{1}{V} - \sqrt{\frac{(1+V)(1-3V)}{V(1-V)}}$$

Square root singularity expansion near dominant singularity

$$V = \frac{1}{3} - cte\sqrt{(1-\frac{t}{t_c})} + O(1-\frac{t}{t_c})$$

with $t_c = \frac{4}{27}$

$$f = \frac{1}{4} - \frac{3}{4}(1-\frac{t}{t_c}) + \frac{\sqrt{3}}{2}(1-\frac{t}{t_c})^{\frac{3}{2}} + O((1-\frac{t}{t_c})^2)$$

$$A = 3 - \sqrt{cte \cdot \sqrt{1-\frac{t}{t_c}} + O(\sqrt{1-\frac{t}{t_c}})} = 3 - cte \cdot (1-\frac{t}{t_c})^{\frac{1}{4}} + O((1-\frac{t}{t_c})^{\frac{1}{2}})$$

The area generating function

We have the singular expansions:

$$f = \frac{1}{4} - \frac{3}{4}(1 - \frac{t}{t_c}) + \frac{\sqrt{3}}{2}(1 - \frac{t}{t_c})^{\frac{3}{2}} + O((1 - \frac{t}{t_c})^2)$$

$$A = 3 - cte \cdot (1 - \frac{t}{t_c})^{\frac{1}{4}} + O((1 - \frac{t}{t_c})^{\frac{1}{2}})$$

From transfert theorems: $g \sim (1 - \frac{t}{t_c})^{\alpha} \Rightarrow [t^n]g \sim \frac{n^{-1-\alpha}}{\Gamma(-\alpha)} t_c^{-n}$
we obtain:

$$[t^n]A \underset{n\to\infty}{\sim} cte \cdot n^{-\frac{5}{4}} t_c^{-n}$$

$$[t^n]f \underset{n\to\infty}{\sim} cte \cdot n^{-\frac{5}{2}} t_c^{-n}$$

Then the average area is

$$\frac{[t^n]A}{[t^n]f} \underset{n\to\infty}{\sim} cte \cdot n^{\frac{5}{4}}$$