From order one catalytic decompositions to context-free specifications, bijectively

Gilles Schaeffer
LIX, CNRS, Institut Polytechnique de Paris
join work with Enrica Duchi

FPSAC 2025
July 21, 2025, Sapporo

CONTEXT-FREE SPECIFICATIONS

Enumerative combinatorics and generating functions

Let $\mathcal{A}$ be a set of combinatorial objects equipped with an integer size $|.|$ and assume that for each n the set

$$\mathcal{A}_n = \{a \in \mathcal{A} \text{ s.t. } |a| = n\}$$

is finite, and let $a_n = |\mathcal{A}_n|$ denote its cardinality.

The generating function (gf) of the class $\mathcal{A}$ w.r.t. the size is

$$A \equiv A(t) := \sum_{n \geq 0} a_n t^n = \sum_{\alpha \in \mathcal{A}} t^{|\alpha|}$$

Refined enumeration:

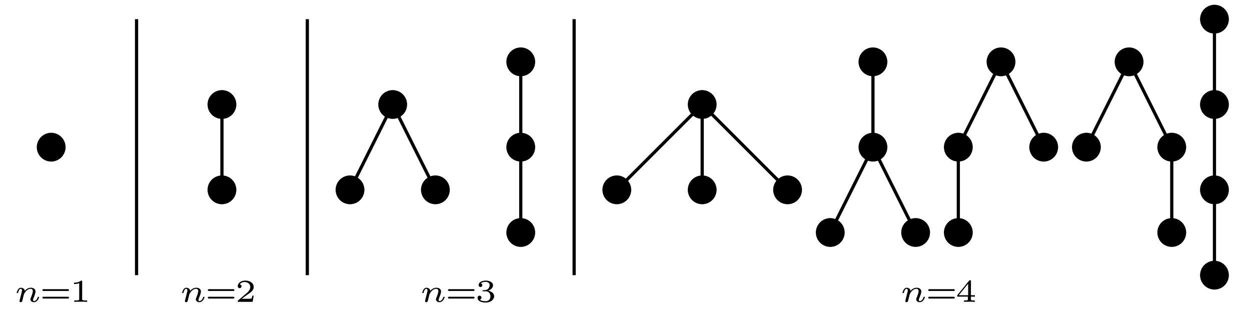
$$A(u) \equiv A(u,t) := \sum_{n,k \geq 0} a_{k,n} u^k t^n = \sum_{\alpha \in \mathcal{A}} u^{p(\alpha)} t^{|\alpha|}$$

for some parameter $p : \mathcal{A} \to \mathbb{Z}$, and $a_{k,n} = |\{a \in \mathcal{A}_n \mid p(a) = k\}$

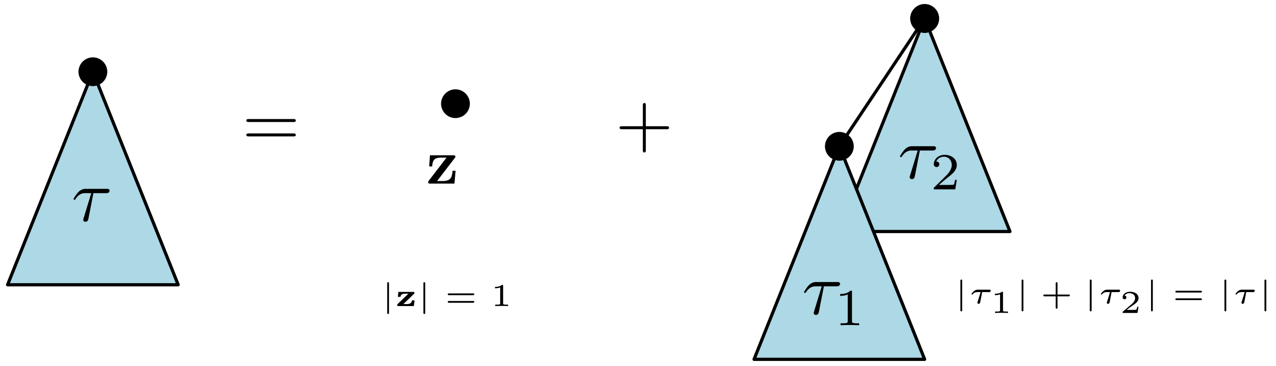
Plane trees

Plane trees (aka ordered trees)

$\mathcal{A}_n = \{$plane trees with n vertices$\}$



Characterized by their decomposition at root edge

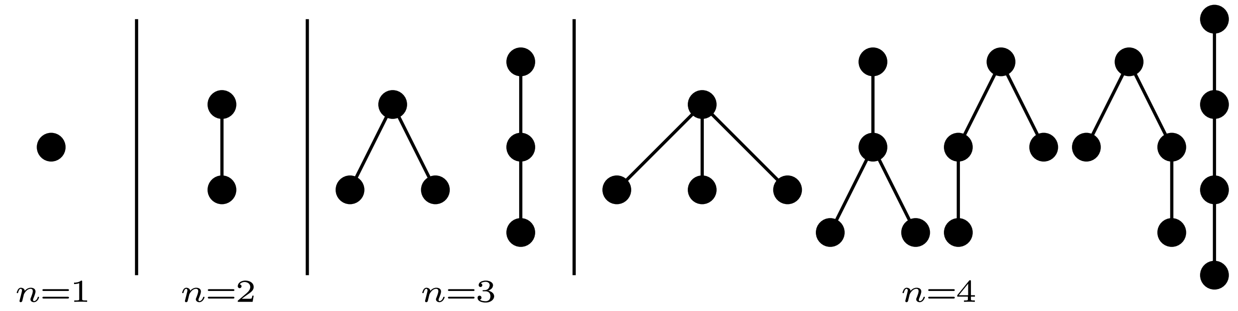


a size preserving recursive bijection

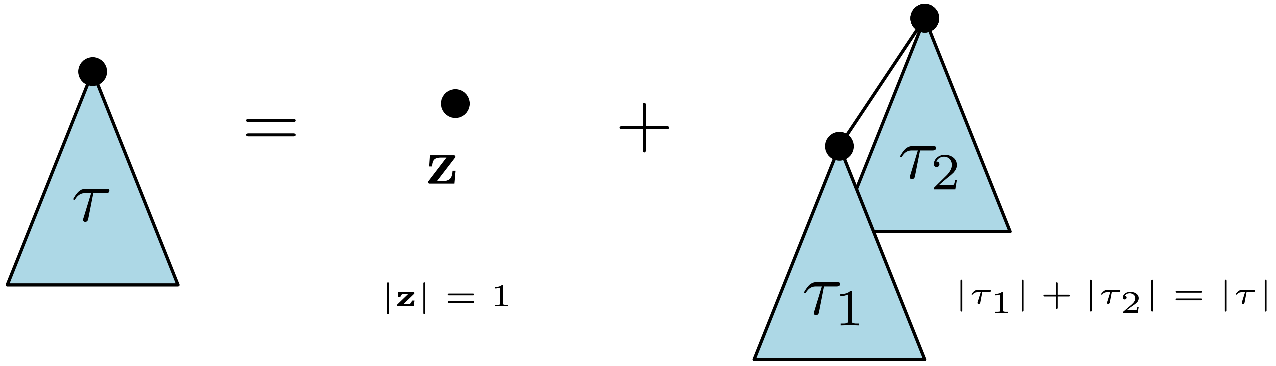
Plane trees

Plane trees (aka ordered trees)

$\mathcal{A}_n = \{$plane trees with n vertices$\}$



Characterized by their decomposition at root edge



$\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

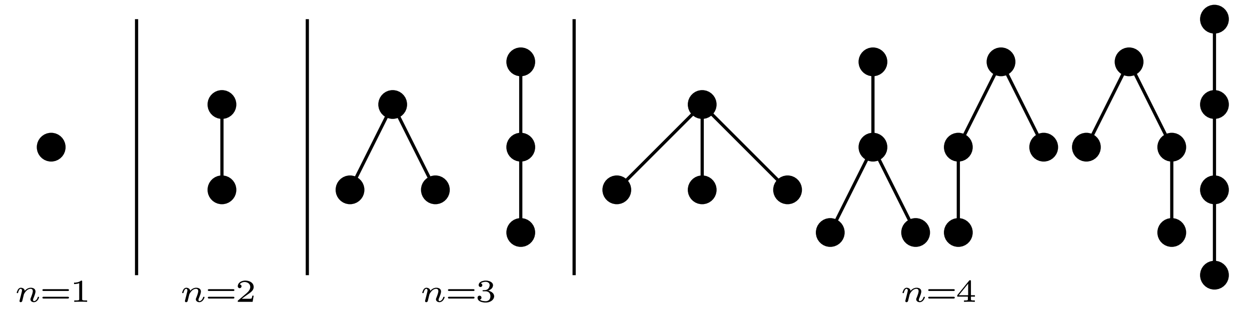
a size preserving recursive bijection

aka a symbolic specification of the class of plane trees

Plane trees

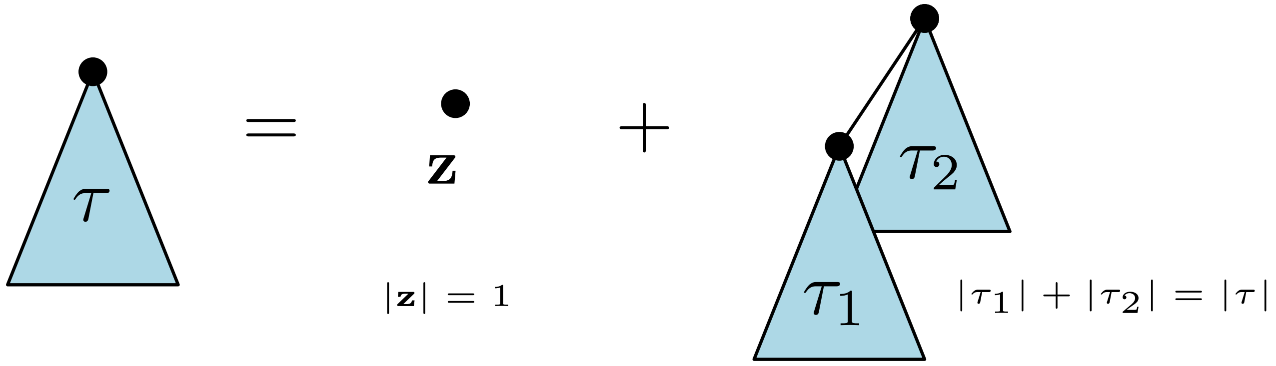
Plane trees (aka ordered trees)

$\mathcal{A}_n = \{\text{plane trees with } n \text{ vertices}\}$



Characterized by their decomposition at root edge

$A(t) = t + t^2 + 2t^3 + 5t^4 + O(t^5)$



$\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

a size preserving recursive bijection

aka a symbolic specification of the class of plane trees

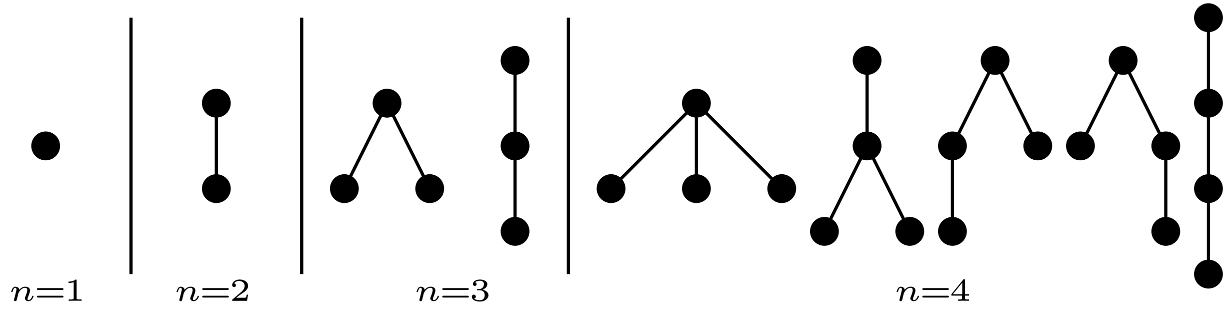
The gf translation:

$$A(t) = \sum_{\tau\in\mathcal{A}} t^{|\tau|} = t^1 + \sum_{(\tau_1,\tau_2)\in\mathcal{A}\times\mathcal{A}} t^{|\tau_1|+|\tau_2|} = t + \sum_{\tau_1\in\mathcal{A}} t^{|\tau_1|} \sum_{\tau_2\in\mathcal{A}} t^{|\tau_2|} = t + A(t)^2$$

Plane trees

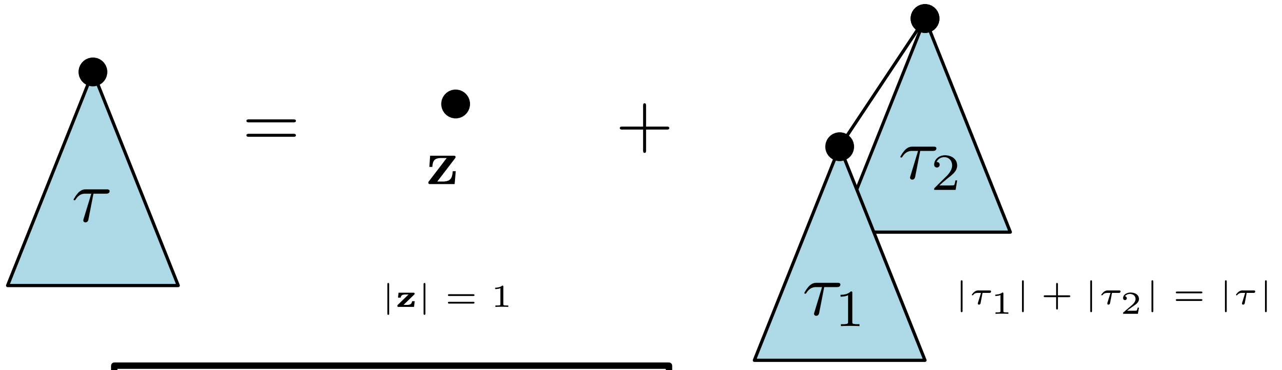
Plane trees (aka ordered trees)

$\mathcal{A}_n = \{\text{plane trees with } n \text{ vertices}\}$



Characterized by their decomposition at root edge

$A(t) = t + t^2 + 2t^3 + 5t^4 + O(t^5)$



$\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

a size preserving recursive bijection

aka a symbolic specification of the class of plane trees

The gf translation:

$$A(t) = \sum_{\tau \in \mathcal{A}} t^{|\tau|} = t^1 + \sum_{(\tau_1,\tau_2) \in \mathcal{A}\times\mathcal{A}} t^{|\tau_1|+|\tau_2|} = t + \sum_{\tau_1 \in \mathcal{A}} t^{|\tau_1|} \sum_{\tau_2 \in \mathcal{A}} t^{|\tau_2|} = t + A(t)^2$$

$A(t) = t + A(t)^2$

Plane trees

A symbolic specification

$$\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$$ with $\mathbf{z}$ atom of size 1 and additive size

The gf translation

$$A(t) = t + A(t)^2$$ with unique sol $A(t) = \sum_{n\geq 0} a_n t^n$ in $\mathbb{C}[[t]]$.

Context free languages and algebraic specifications/decompositions

More generally we like particularly well funded **context-free specifications**:

$$\begin{cases} \mathcal{F}^{(1)} & \equiv & \mathcal{P}^{(1)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \\ & \vdots & \\ \mathcal{F}^{(k)} & \equiv & \mathcal{P}^{(k)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \end{cases}$$

with each $\mathcal{P}^{(i)}$ a finite combination of $+$ and $\times$ operators

e.g. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

Context free languages and algebraic specifications/decompositions

More generally we like particularly well funded **context-free specifications**:

$$\left\{ \begin{array}{rcl} \mathcal{F}^{(1)} & \equiv & \mathcal{P}^{(1)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \\ & \vdots & \\ \mathcal{F}^{(k)} & \equiv & \mathcal{P}^{(k)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \end{array} \right.$$

with each $\mathcal{P}^{(i)}$ a finite combination of $+$ and $\times$ operators

e.g. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

as their gf translation is an $\mathbb{N}$-algebraic system:

$$\left\{ \begin{array}{rcl} F^{(1)} & = & P^{(1)}(t; F^{(1)}, \ldots, F^{(k)}) \\ & \vdots & \\ F^{(k)} & = & P^{(k)}(t; F^{(1)}, \ldots, F^{(k)}) \end{array} \right.$$

with each $P^{(i)}$ a polynomial with non negative coefficients, and with a unique power series solution $F^{(1)} \equiv F^{(1)}(t) = \sum_{n \geq 0} F_n^{(1)} t^n$ in $\mathbb{C}[[t]]$.

e.g. $A(t) = t + A(t)^2$

Context free languages and algebraic specifications/decompositions

More generally we like particularly well funded **context-free specifications**:

$$\begin{cases} \mathcal{F}^{(1)} & \equiv & \mathcal{P}^{(1)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \\ & \vdots & \\ \mathcal{F}^{(k)} & \equiv & \mathcal{P}^{(k)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \end{cases}$$

with each $\mathcal{P}^{(i)}$ a finite combination of $+$ and $\times$ operators

e.g. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

as their gf translation is an $\mathbb{N}$-algebraic system:

$$\begin{cases} F^{(1)} & = & P^{(1)}(t; F^{(1)}, \ldots, F^{(k)}) \\ & \vdots & \\ F^{(k)} & = & P^{(k)}(t; F^{(1)}, \ldots, F^{(k)}) \end{cases}$$

with each $P^{(i)}$ a polynomial with non negative coefficients, and with a unique power series solution $F^{(1)} \equiv F^{(1)}(t) = \sum_{n \geq 0} F_n^{(1)} t^n$ in $\mathbb{C}[[t]]$.

e.g. $A(t) = t + A(t)^2$

Combinatorial structures that admit such a context-free specification are **tamed**...

⇒ **exact formulas or efficient enumeration algorithms**

⇒ **asymptotic enumeration via singularity analysis**

⇒ **linear time uniform random generation algorithms**

Context free languages and algebraic specifications/decompositions

More generally we like particularly well funded **context-free specifications**:

$$\begin{cases} \mathcal{F}^{(1)} & \equiv & \mathcal{P}^{(1)}(\mathbf{z}; \mathcal{F}^{(1)}, \dots, \mathcal{F}^{(k)}) \\ & \vdots & \\ \mathcal{F}^{(k)} & \equiv & \mathcal{P}^{(k)}(\mathbf{z}; \mathcal{F}^{(1)}, \dots, \mathcal{F}^{(k)}) \end{cases}$$

with each $\mathcal{P}^{(i)}$ a finite combination of $+$ and $\times$ operators

e.g. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

as their gf translation is an $\mathbb{N}$-algebraic system:

$$\begin{cases} F^{(1)} & = & P^{(1)}(t; F^{(1)}, \dots, F^{(k)}) \\ & \vdots & \\ F^{(k)} & = & P^{(k)}(t; F^{(1)}, \dots, F^{(k)}) \end{cases}$$

with each $P^{(i)}$ a polynomial with non negative coefficients, and with a unique power series solution $F^{(1)} \equiv F^{(1)}(t) = \sum_{n \geq 0} F_n^{(1)} t^n$ in $\mathbb{C}[[t]]$.

e.g. $A(t) = t + A(t)^2$

Applies in particular to non ambiguous context free grammars.

(Chomsky-Schützenberger theorem)

Context free languages and algebraic specifications/decompositions

More generally we like particularly well funded **context-free specifications**:

$$\begin{cases} \mathcal{F}^{(1)} & \equiv & \mathcal{P}^{(1)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \\ & \vdots & \\ \mathcal{F}^{(k)} & \equiv & \mathcal{P}^{(k)}(\mathbf{z}; \mathcal{F}^{(1)}, \ldots, \mathcal{F}^{(k)}) \end{cases}$$

with each $\mathcal{P}^{(i)}$ a finite combination of $+$ and $\times$ operators

e.g. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

as their gf translation is an $\mathbb{N}$-algebraic system:

$$\begin{cases} F^{(1)} & = & P^{(1)}(t; F^{(1)}, \ldots, F^{(k)}) \\ & \vdots & \\ F^{(k)} & = & P^{(k)}(t; F^{(1)}, \ldots, F^{(k)}) \end{cases}$$

with each $P^{(i)}$ a polynomial with non negative coefficients, and with a unique power series solution $F^{(1)} \equiv F^{(1)}(t) = \sum_{n \geq 0} F_n^{(1)} t^n$ in $\mathbb{C}[[t]]$.

e.g. $A(t) = t + A(t)^2$

Applies in particular to non ambiguous context free grammars.

(Chomsky-Schützenberger theorem)

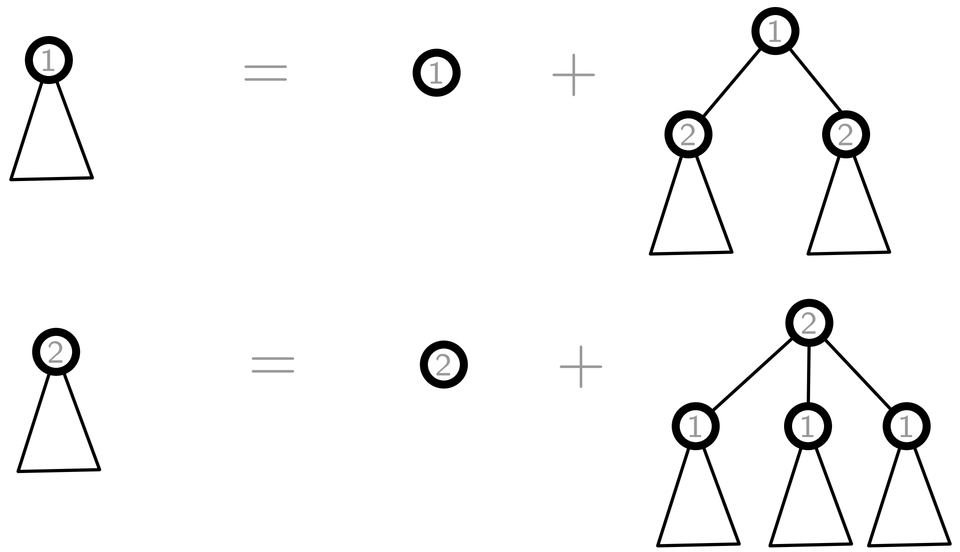
Conversely when the gf of a combinatorial familly $\mathcal{A}$ is known to be $\mathbb{N}$-algebraic, one would like to explain it via an encoding by words of a context-free grammar.

(Schützenberger's methodology for algebraic gf)

Context-free specifications and multitype simply generated trees

Context-free decompositions are naturally associated with multitype simply generated trees:

$$\begin{cases} \mathcal{F}^{(1)} \equiv \mathbf{z} + \mathbf{z} \times \mathcal{F}^{(2)} \times \mathcal{F}^{(2)} \\ \mathcal{F}^{(2)} \equiv \mathbf{z} + \mathbf{z} \times \mathcal{F}^{(1)} \times \mathcal{F}^{(1)} \times \mathcal{F}^{(1)} \end{cases} \Leftrightarrow$$

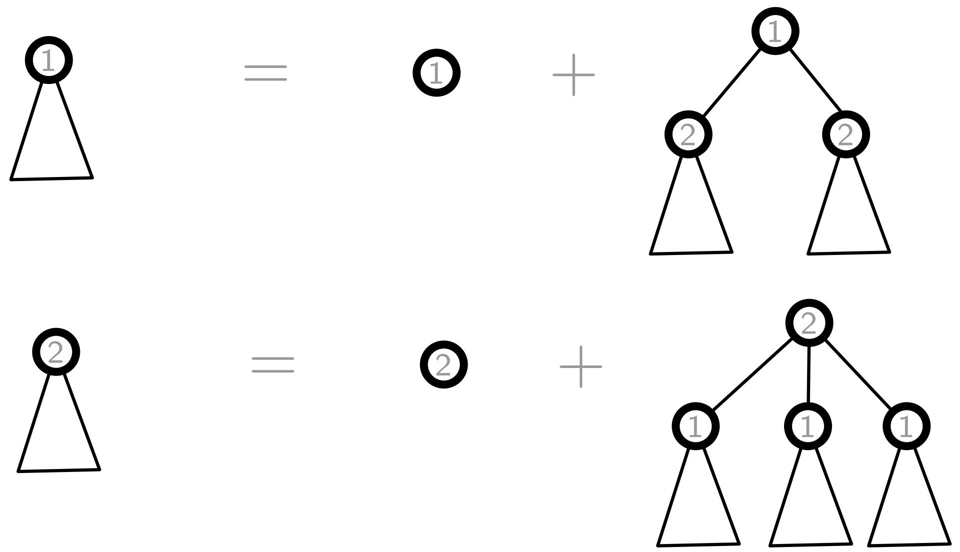


The *derivation trees of a context-free specification* are multitype simply generated trees, *i.e.* trees specified by the allowed node progeny for each color, with independent subtrees.

Context-free specifications and multitype simply generated trees

Context-free decompositions are naturally associated with multitype simply generated trees:

$$\begin{cases} \mathcal{F}^{(1)} \equiv \mathbf{z} + \mathbf{z} \times \mathcal{F}^{(2)} \times \mathcal{F}^{(2)} \\ \mathcal{F}^{(2)} \equiv \mathbf{z} + \mathbf{z} \times \mathcal{F}^{(1)} \times \mathcal{F}^{(1)} \times \mathcal{F}^{(1)} \end{cases} \Leftrightarrow$$



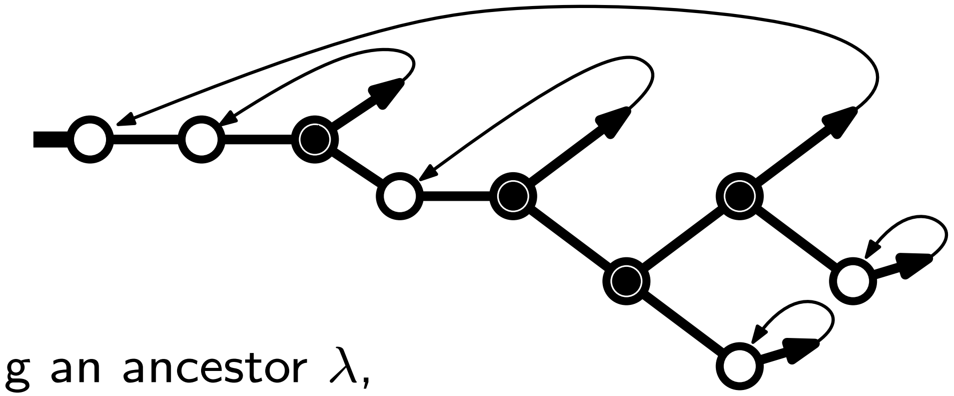
The *derivation trees of a context-free specification* are multitype simply generated trees, *i.e.* trees specified by the allowed node progeny for each color, with independent subtrees.

Conversely when the gf of a combinatorial familly $\mathcal{A}$ is known to be $\mathbb{N}$-algebraic, one would like to explain it via a **context-free specification** of $\mathcal{A}$ or via a **bijection with trees**.

(Standard reformulation of Schützenberger's methodology)

CATALYTIC DECOMPOSITIONS

The example of planar λ-terms

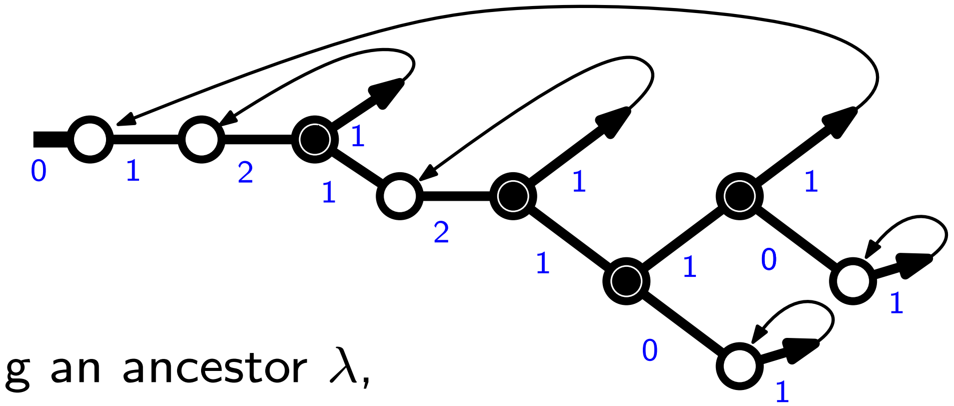


Planar λ-terms can be presented as trees with

- **applications**: binary nodes
- λ-**abstractions**: unary nodes
- **variables**: leaves, represented as arrows, each matching an ancestor λ,

with condition that each λ is binded to exactly one variable in a planar way...

The example of planar λ-terms



Planar λ-terms can be presented as trees with

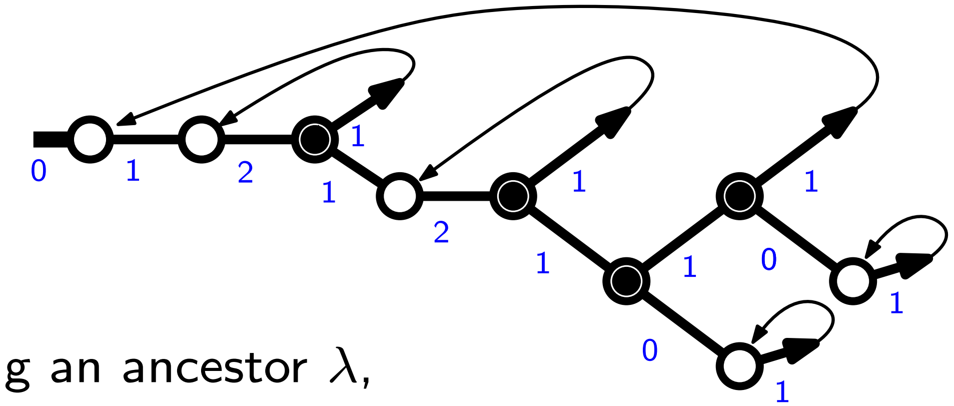
- **applications**: binary nodes
- λ-**abstractions**: unary nodes
- **variables**: leaves, represented as arrows, each matching an ancestor λ,

with condition that each λ is binded to exactly one variable in a planar way...

Equivalently, in each subterm there are more variables than abstractions,

or the *catalytic parameter*, $\text{excess}(\tau) = \#\{\text{variables}\} - \#\{\text{abstractions}\}$, is non negative everywhere.

The example of planar λ-terms



Planar λ-terms can be presented as trees with

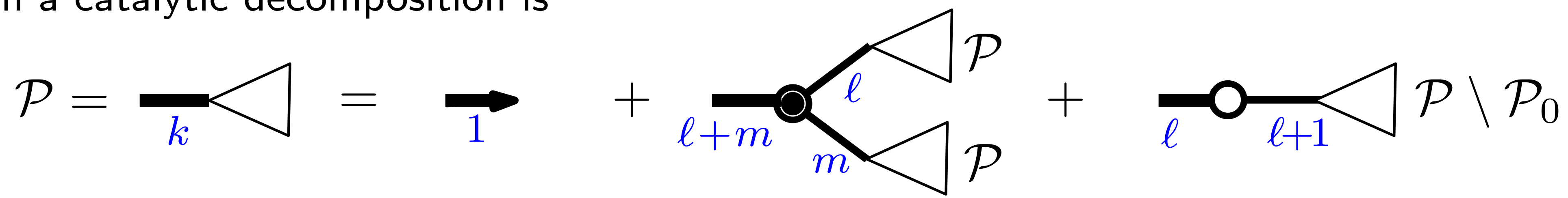
- **applications**: binary nodes
- **λ-abstractions**: unary nodes
- **variables**: leaves, represented as arrows, each matching an ancestor λ,

with condition that each λ is binded to exactly one variable in a planar way...

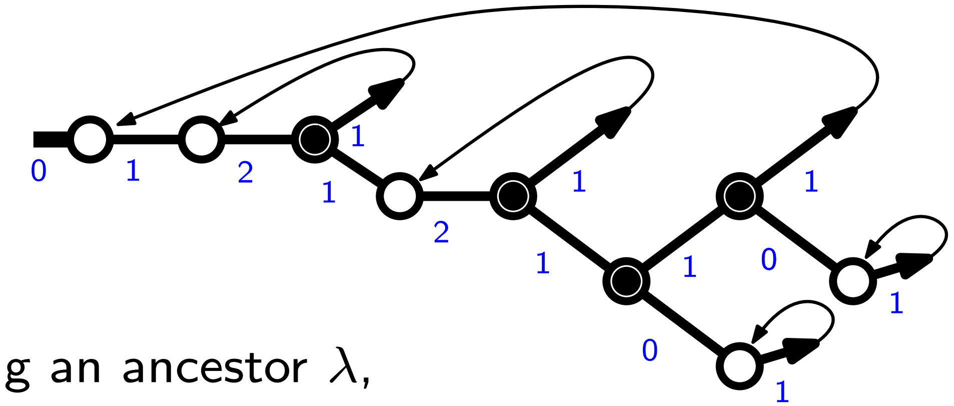
Equivalently, in each subterm there are more variables than abstractions,

or the *catalytic parameter*, excess$(\tau) = \#\{\text{variables}\} - \#\{\text{abstractions}\}$, is non negative everywhere.

Then a catalytic decomposition is



The example of planar λ-terms



Planar λ-terms can be presented as trees with

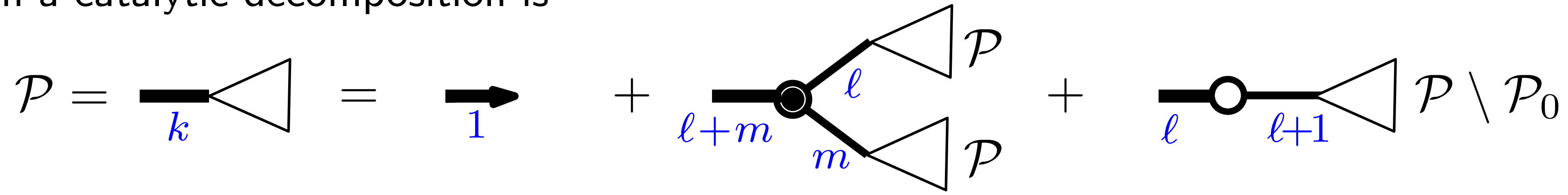
- **applications**: binary nodes
- **λ-abstractions**: unary nodes
- **variables**: leaves, represented as arrows, each matching an ancestor λ,

with condition that each λ is binded to exactly one variable in a planar way...

Equivalently, in each subterm there are more variables than abstractions,

or the *catalytic parameter*, excess$(\tau) = \#\{\text{variables}\} - \#\{\text{abstractions}\}$, is non negative everywhere.

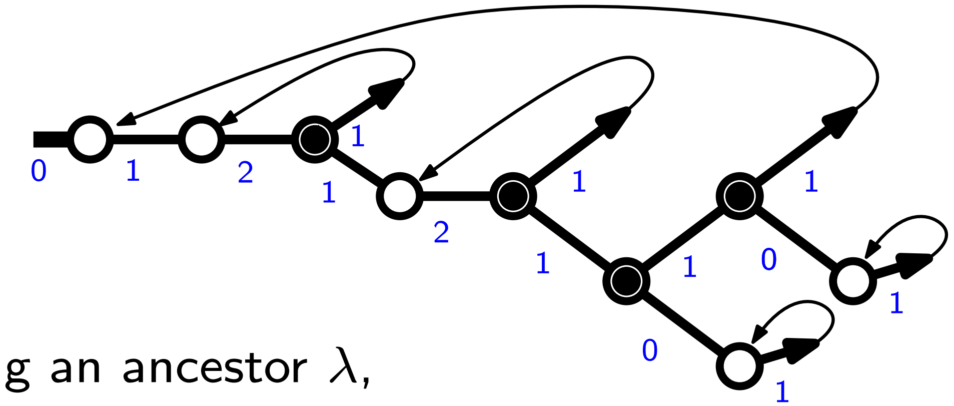
Then a catalytic decomposition is



and the catalytic equation for the gf $P(u) = \sum_{\tau \in \mathcal{P}} t^{|\tau|} u^{excess(\tau)}$ is

$$P(u) = tu + tP(u)^2 + \frac{t}{u}(P(u) - P(0))$$

The example of planar λ-terms



Planar λ-terms can be presented as trees with

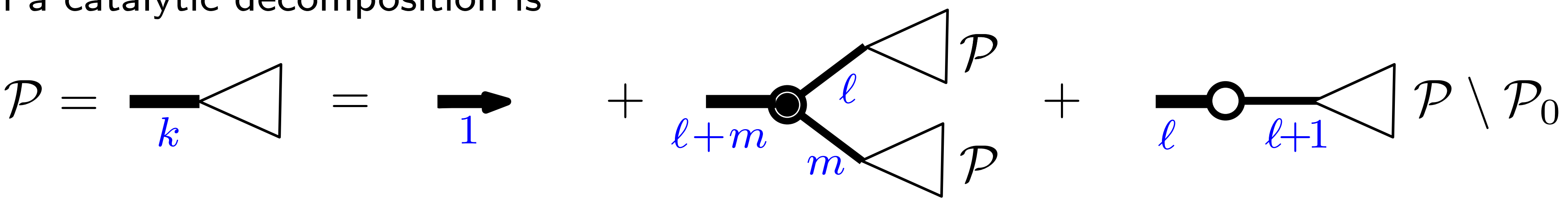
- **applications**: binary nodes
- **λ-abstractions**: unary nodes
- **variables**: leaves, represented as arrows, each matching an ancestor λ,

with condition that each λ is binded to exactly one variable in a planar way...

Equivalently, in each subterm there are more variables than abstractions,

or the *catalytic parameter*, excess$(\tau) = \#\{\text{variables}\} - \#\{\text{abstractions}\}$, is non negative everywhere.

Then a catalytic decomposition is



and the catalytic equation for the gf $P(u) = \sum_{\tau \in \mathcal{P}} t^{|\tau|} u^{excess(\tau)}$ is

$$P(u) \quad = \quad tu \quad + \quad tP(u)^2 \quad + \quad \frac{t}{u}(P(u) - P(0))$$

This equation is not algebraic, the decomposition is not context free.

1-variable catalytic equations

The equation $P(u) = t(u + P(u)^2 + \frac{1}{u}(P(u) - P(0))$ is a special case of 1-variable catalytic equation,

$$Q\left(F(u), f_1, f_2, \ldots, f_k, u, t\right) = 0$$

where Q is a polynomial with coefficients in some field $\mathbb{F}$
and we seek the unknown formal power series $F(u) \equiv F(t,u) \in \mathbb{F}[[t,u]]$ and $f_i \equiv f_i(t) \in \mathbb{F}[[t]]$.

These equations also surface in various other enumeration problems, for instance for

- Families of pattern avoiding permutations (Zeilberger 92, Bona, Bousquet-Mélou, late 90's)
- Families of Tamari intervals (Chapoton, 2000's, Bousquet-Mélou-Chapoton 2022)
- Families of Planar (normal) λ-terms (Zeilberger and Giorgietti, 2015)
- Fighting fish and variants (Duchi et al, 2016)
- Fully parked trees (Chen 2021, Contat et al 2023)
- ...

1-variable catalytic equations

The equation $P(u) = t(u + P(u)^2 + \frac{1}{u}(P(u) - P(0))$ is a special case of 1-variable catalytic equation,

$$Q\left(F(u), f_1, f_2, \ldots, f_k, u, t\right) = 0$$

where Q is a polynomial with coefficients in some field $\mathbb{F}$
and we seek the unknown formal power series $F(u) \equiv F(t,u) \in \mathbb{F}[[t,u]]$ and $f_i \equiv f_i(t) \in \mathbb{F}[[t]]$.

The celebrated **Bousquet-Mélou – Jehanne theorem** states that 1-variable catalytic equations of the form

$$F(u) = F_0(u) + tQ(F(u), \Delta F(u), \ldots, \Delta^k F(u), u, t)$$

where $F_0(u)$ and $Q(v, w_1, \ldots, w_k, u)$ are polynomials with coefficients in $\mathbb{F}$, and

$$\Delta^k F(u) = \frac{F(u) - f_1 - uf_2 - \ldots - u^{k-1} f_k}{u^k},$$

have unique solutions, and it provides a non degenerated system of algebraic equations that they satisfy.

Explicit BMJ theorem for order one 1-catalytic equations

Let $Q(v, w, u)$ be a polynomial with $Q(0, 0, u) \neq 0$

and $F(u) \equiv F(t, u)$ the unique fps solution of the catalytic equation

$$F(u) = t\, Q\left(F(u), \frac{1}{u}(F(u) - f), u\right), \qquad \text{where } f \equiv f(t) = F(t, 0).$$

Let U, V, W and R be the unique fps satisfying the system

$$\begin{cases} V &= t \cdot Q(V, W, U) \\ R &= t \cdot (1 + R) \cdot Q'_v(V, W, U) \\ U &= t \cdot (1 + R) \cdot Q'_w(V, W, U) \\ W &= t \cdot (1 + R) \cdot Q'_u(V, W, U) \end{cases}$$

Then f is given by $\quad f = V - UW \quad$ or $\quad t f'_t = (1 + R) \cdot V$

Explicit BMJ theorem for order one 1-catalytic equations

Let $Q(v,w,u)$ be a polynomial with $Q(0,0,u) \neq 0$

and $F(u) \equiv F(t,u)$ the unique fps solution of the catalytic equation

$$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u)-f), u\right), \qquad \text{where } f \equiv f(t) = F(t,0).$$

$$P(u) = t(u + P(u)^2 + \tfrac{1}{u}(P(u) - P(0))$$

Let U, V, W and R be the unique fps satisfying the system

$$\begin{cases} V &= t \cdot Q(V,W,U) \\ R &= t \cdot (1+R) \cdot Q'_v(V,W,U) \\ U &= t \cdot (1+R) \cdot Q'_w(V,W,U) \\ W &= t \cdot (1+R) \cdot Q'_u(V,W,U) \end{cases}$$

$$\begin{cases} V &= t \cdot (U + V^2 + W) \\ R &= t \cdot (1+R) \cdot 2V \\ U &= t \cdot (1+R) \\ W &= t \cdot (1+R) \end{cases}$$

$$P(0) = V - UW$$

Then f is given by $\quad f = V - UW \quad$ or $\quad t f'_t = (1+R) \cdot V$

Explicit BMJ theorem for order one 1-catalytic equations

Let $Q(v, w, u)$ be a polynomial with $Q(0, 0, u) \neq 0$

and $F(u) \equiv F(t, u)$ the unique fps solution of the catalytic equation

$$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u) - f), u\right), \qquad \text{where } f \equiv f(t) = F(t, 0).$$

$$P(u) = t(u + P(u)^2 + \tfrac{1}{u}(P(u) - P(0))$$

Let U, V, W and R be the unique fps satisfying the system

$$\begin{cases} V & = & t \cdot Q(V, W, U) \\ R & = & t \cdot (1 + R) \cdot Q'_v(V, W, U) \\ U & = & t \cdot (1 + R) \cdot Q'_w(V, W, U) \\ W & = & t \cdot (1 + R) \cdot Q'_u(V, W, U) \end{cases}$$

$$\begin{cases} V & = & t \cdot (U + V^2 + W) \\ R & = & t \cdot (1 + R) \cdot 2V \\ U & = & t \cdot (1 + R) \\ W & = & t \cdot (1 + R) \end{cases}$$

$$P(0) = V - UW$$

Then f is given by $\qquad f = V - UW \qquad$ or $\qquad t f'_t = (1 + R) \cdot V$

$\Rightarrow$ The particularly simple form of this parametrization calls for a combinatorial lifting.

Explicit BMJ theorem for order one 1-catalytic equations

Let $Q(v, w, u)$ be a polynomial with $Q(0, 0, u) \neq 0$

and $F(u) \equiv F(t, u)$ the unique fps solution of the catalytic equation

$$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u) - f), u\right), \qquad \text{where } f \equiv f(t) = F(t, 0).$$

$$P(u) = t(u + P(u)^2 + \tfrac{1}{u}(P(u) - P(0))$$

Let U, V, W and R be the unique fps satisfying the system

$$\begin{cases} V &= t \cdot Q(V, W, U) \\ R &= t \cdot (1 + R) \cdot Q'_v(V, W, U) \\ U &= t \cdot (1 + R) \cdot Q'_w(V, W, U) \\ W &= t \cdot (1 + R) \cdot Q'_u(V, W, U) \end{cases}$$

$$\begin{cases} V &= t \cdot (U + V^2 + W) \\ R &= t \cdot (1 + R) \cdot 2V \\ U &= t \cdot (1 + R) \\ W &= t \cdot (1 + R) \end{cases}$$

$$P(0) = V - UW$$

Then f is given by $\quad f = V - UW \quad$ or $\quad t f'_t = (1 + R) \cdot V$

$\Rightarrow$ The particularly simple form of this parametrization calls for a combinatorial lifting.

$\Rightarrow$ When Q is a polynomial with integer coefficients, the system is $\mathbb{N}$-algebraic !

Explicit BMJ theorem for order one 1-catalytic equations

Let $Q(v,w,u)$ be a polynomial with $Q(0,0,u) \neq 0$

and $F(u) \equiv F(t,u)$ the unique fps solution of the catalytic equation

$$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u)-f), u\right), \qquad \text{where } f \equiv f(t) = F(t,0).$$

$$P(u) = t(u + P(u)^2 + \tfrac{1}{u}(P(u) - P(0))$$

Let U, V, W and R be the unique fps satisfying the system

$$\begin{cases} V &= t \cdot Q(V,W,U) \\ R &= t \cdot (1+R) \cdot Q'_v(V,W,U) \\ U &= t \cdot (1+R) \cdot Q'_w(V,W,U) \\ W &= t \cdot (1+R) \cdot Q'_u(V,W,U) \end{cases}$$

$$\begin{cases} V &= t \cdot (U + V^2 + W) \\ R &= t \cdot (1+R) \cdot 2V \\ U &= t \cdot (1+R) \\ W &= t \cdot (1+R) \end{cases}$$

$$P(0) = V - UW$$

Then f is given by $\quad f = V - UW \quad$ or $\quad t f'_t = (1+R) \cdot V$

$\Rightarrow$ The particularly simple form of this parametrization calls for a combinatorial lifting.

$\Rightarrow$ When Q is a polynomial with integer coefficients, the system is $\mathbb{N}$-algebraic !

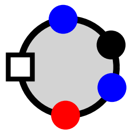
$\Rightarrow$ explain it via a **bijection with some simply generated trees**.

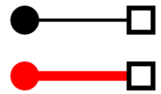
A MODEL FOR CATALYTIC EQUATIONS

Decorated trees and non negative trees

non-negative $\mathcal{Q}$-tree $=$ necklace tree s.t.

the excess at each pearl is non negative.

$\mathcal{Q} = \{$  $, \ldots \}$



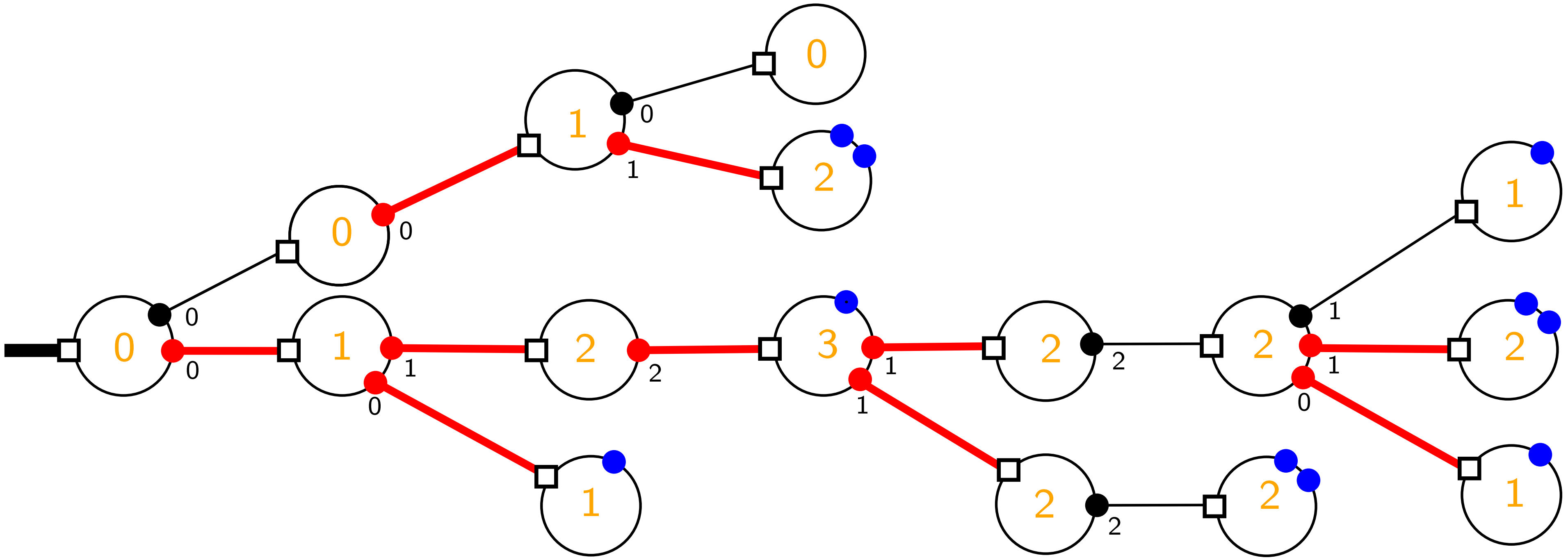
●, ● are all matched.

#{●} ≥ #{●}
in planted subtrees

Observe:

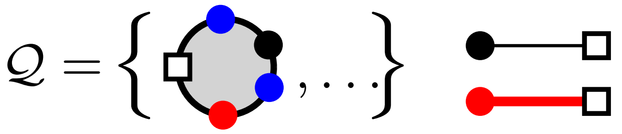
slightly stronger condition than
just asking non negative excess on vertices

excess $=$ #{●} $-$ #{●}



Non negative $\mathcal{Q}$-trees and catalytic equations

Let $\mathcal{F} = \{$ **non-negative $\mathcal{Q}$-trees** $\}$,



$\bullet$, ${\color{red}\bullet}$ are all matched.

$\#\{{\color{blue}\bullet}\} \geq \#\{{\color{red}\bullet}\}$ in planted subtrees

$Q(v,w,u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$ the vertex type gf, where q_s are weights

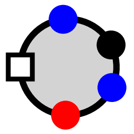
and $F(u) \equiv F(t,u) = \sum_{\tau \in \mathcal{F}} q_\tau t^{|\tau|} u^{\mathrm{excess}(\tau)}$, where $q_\tau = \prod_{s \in \tau} q_s$

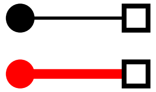
Proposition. The gf $F(u)$ of non negative $\mathcal{Q}$-trees satisfies a catalytic equation of order one:

$$F(u) = tQ\Big(F(u), \tfrac{1}{u}(F(u) - F(0)), u\Big)$$

Non negative $\mathcal{Q}$-trees and catalytic equations

Let $\mathcal{F} = \{$ **non-negative $\mathcal{Q}$-trees** $\}$,

$\mathcal{Q} = \Big\{$  $, \ldots \Big\}$



●, ● are all matched.
$\#\{\bullet\} \geq \#\{\bullet\}$
in planted subtrees

$$Q(v,w,u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{\bullet(s)} u^{\bullet(s)}$$ the vertex type gf, where q_s are weights

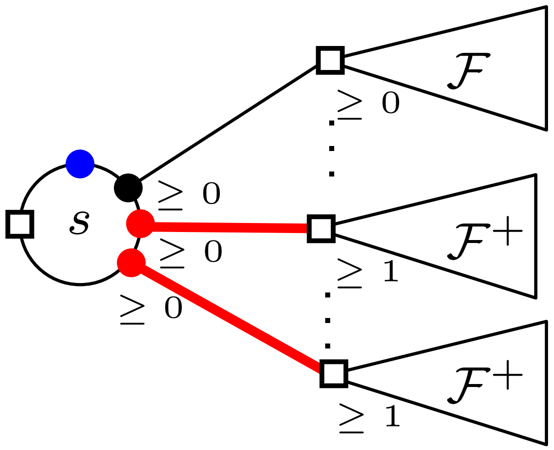
and $F(u) \equiv F(t,u) = \sum_{\tau \in \mathcal{F}} q_\tau t^{|\tau|} u^{\text{excess}(\tau)}$, where $q_\tau = \prod_{s \in \tau} q_s$

Proposition. The gf $F(u)$ of non negative $\mathcal{Q}$-trees satisfies a catalytic equation of order one:

$$F(u) = tQ\Big(F(u), \tfrac{1}{u}(F(u) - F(0)), u\Big)$$

Indeed the equation

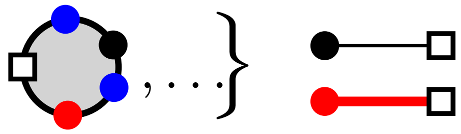
$$F(u) = t \sum_{s \in \mathcal{Q}} q_s F(u)^{\bullet(s)} \left(\tfrac{1}{u}(F(u) - F(0))\right)^{\bullet(s)} u^{\bullet(s)}$$

follows from a decomposition at the root: $\mathcal{F} \equiv \sum_{s \in \mathcal{Q}} q_s \cdot$ 

where $\mathcal{F}^+ = \mathcal{F} \setminus f$

Non negative $\mathcal{Q}$-trees and catalytic equations

Let $\mathcal{F} = \{$ **non-negative $\mathcal{Q}$-trees** $\}$,

$\mathcal{Q} = \{$  $, \ldots \}$

●, ● are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$ in planted subtrees

$Q(v, w, u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{\bullet(s)} u^{\bullet(s)}$ the vertex type gf, where q_s are weights

and $F(u) \equiv F(t, u) = \sum_{\tau \in \mathcal{F}} q_\tau t^{|\tau|} u^{\text{excess}(\tau)}$, where $q_\tau = \prod_{s \in \tau} q_s$

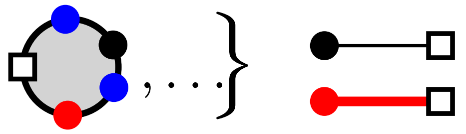
Proposition. The gf $F(u)$ of non negative $\mathcal{Q}$-trees satisfies a catalytic equation of order one:

$$F(u) = tQ\Big(F(u), \tfrac{1}{u}(F(u) - F(0)), u\Big)$$

$\Rightarrow$ non-negative $\mathcal{Q}$-trees give a generic combinatorial interpretation
for catalytic equations of order one with non negative coefficients.

Non negative $\mathcal{Q}$-trees and catalytic equations

Let $\mathcal{F} = \{$ **non-negative $\mathcal{Q}$-trees** $\}$,

$\mathcal{Q} = \{$  $, \ldots \}$

•, • are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$

in planted subtrees

$Q(v, w, u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{\bullet(s)} u^{\bullet(s)}$ the vertex type gf, where q_s are weights

and $F(u) \equiv F(t, u) = \sum_{\tau \in \mathcal{F}} q_\tau t^{|\tau|} u^{\text{excess}(\tau)}$, where $q_\tau = \prod_{s \in \tau} q_s$

Proposition. The gf $F(u)$ of non negative $\mathcal{Q}$-trees satisfies a catalytic equation of order one:

$$F(u) = tQ\Big(F(u), \tfrac{1}{u}(F(u) - F(0)), u\Big)$$

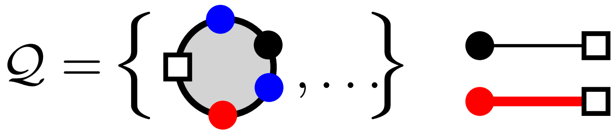
$\Rightarrow$ non-negative $\mathcal{Q}$-trees give a generic combinatorial interpretation
for catalytic equations of order one with non negative coefficients.

Non-negative $\mathcal{Q}$-trees are generic derivation trees for catalytic decompositions.

FROM CATALYTIC DECOMPOSITIONS
TO CONTEXT FREE SPECIFICATIONS

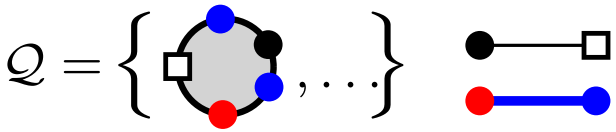
Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree = necklace tree s.t.
necklaces are in $\mathcal{Q}$
the excess at each pearl is non negative.



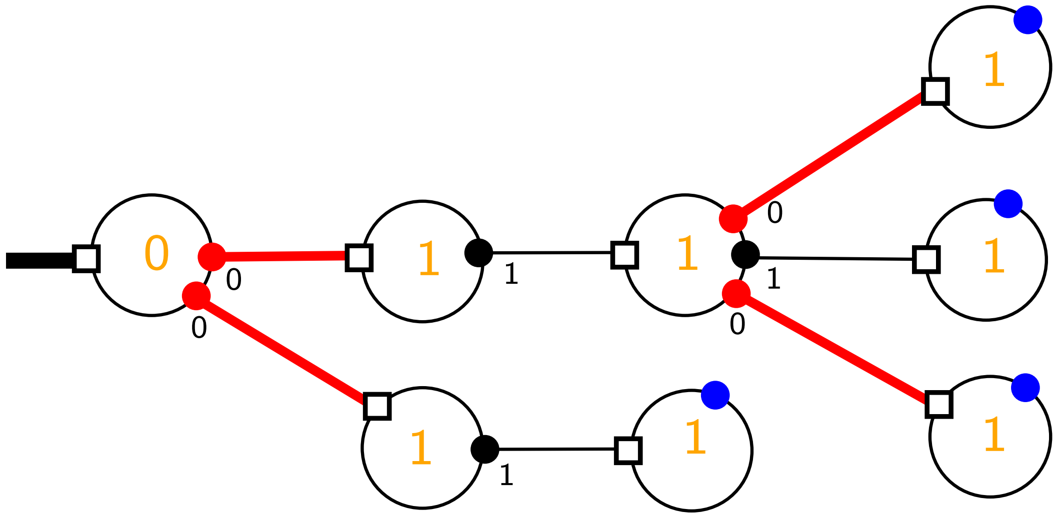
•, • are all matched.
$\#\{\bullet\} \geq \#\{\bullet\}$
in planted subtrees

companion $\mathcal{Q}$- tree = necklace tree s.t.
necklaces are in $\mathcal{Q}$



•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees

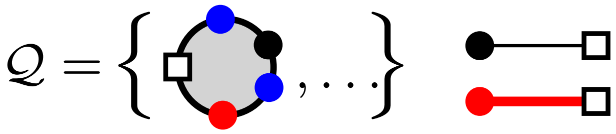


Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree $=$ necklace tree s.t.

necklaces are in $\mathcal{Q}$

the excess at each pearl is non negative.



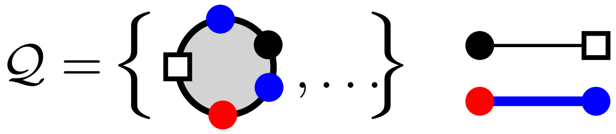
•, • are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$

in planted subtrees

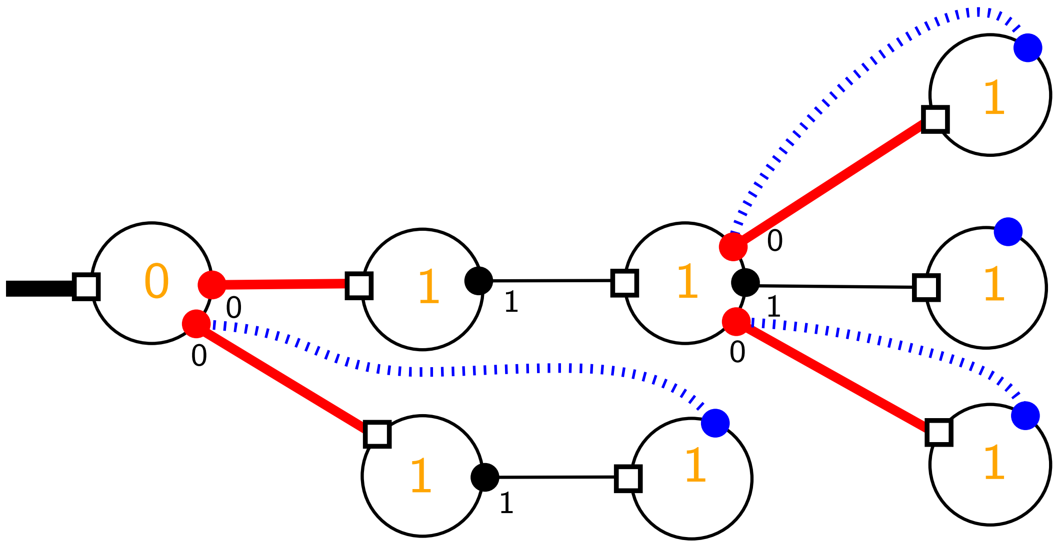
companion $\mathcal{Q}$- tree $=$ necklace tree s.t.

necklaces are in $\mathcal{Q}$



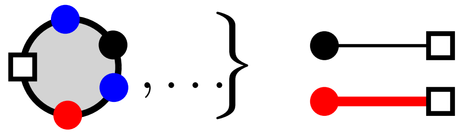
•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees



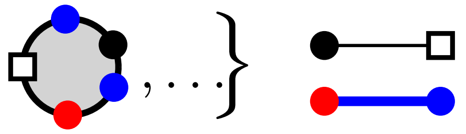
Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree = necklace tree s.t.
necklaces are in $\mathcal{Q}$
the excess at each pearl is non negative.

$\mathcal{Q} = \{$  $, \ldots\}$

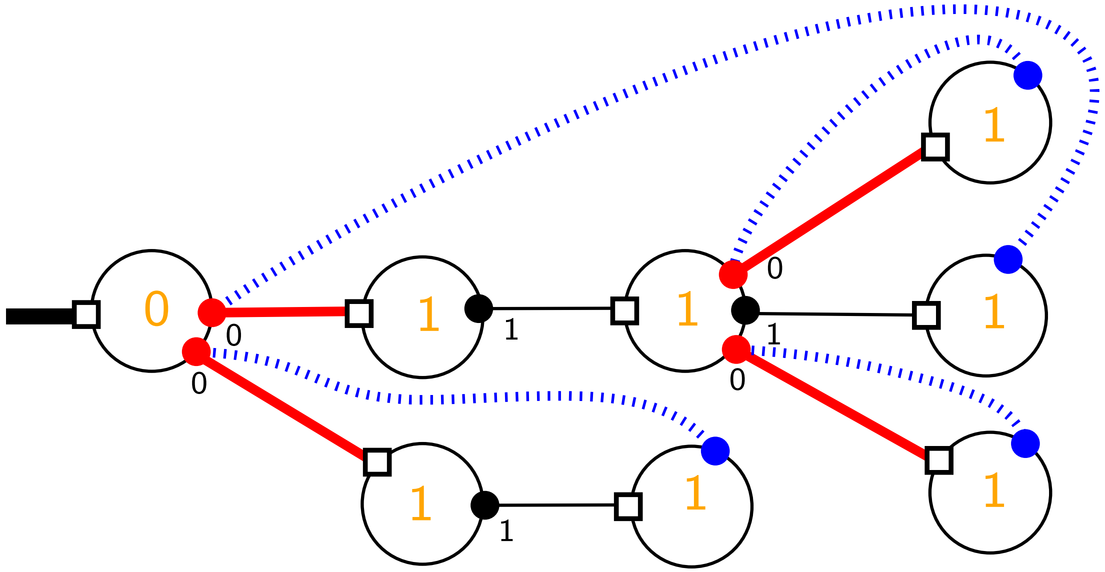
•, • are all matched.
$\#\{\bullet\} \geq \#\{\bullet\}$
in planted subtrees

companion $\mathcal{Q}$- tree = necklace tree s.t.
necklaces are in $\mathcal{Q}$

$\mathcal{Q} = \{$  $, \ldots\}$

•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees

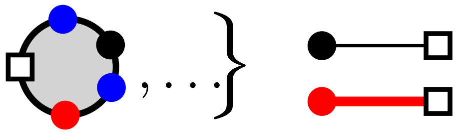


Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$

the excess at each pearl is non negative.

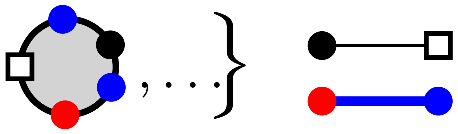
$\mathcal{Q} = \{$  $, \dots \}$

•, • are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$
in planted subtrees

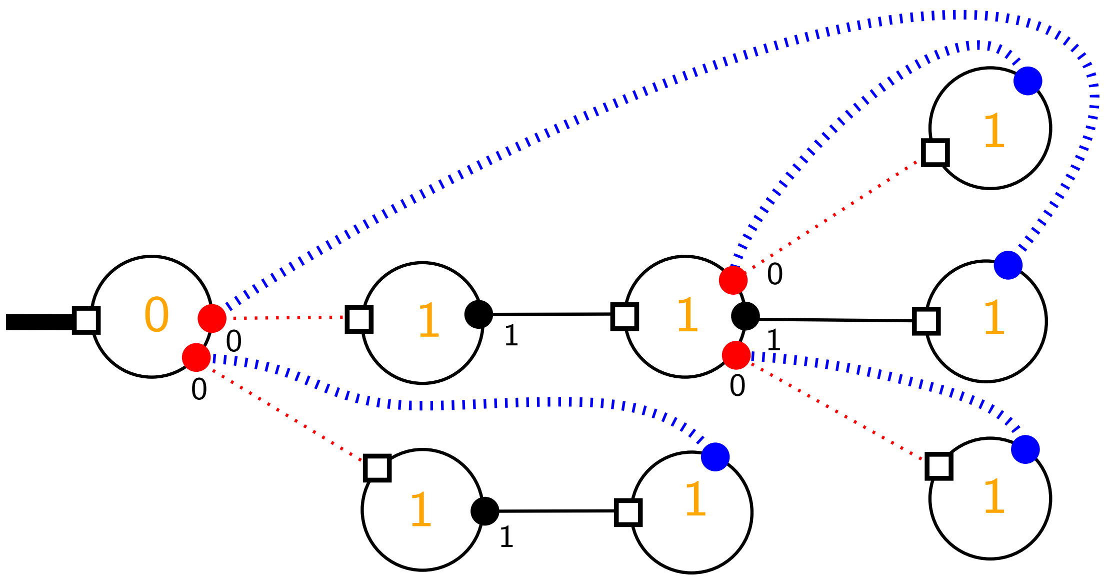
companion $\mathcal{Q}$- tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$

$\mathcal{Q} = \{$  $, \dots \}$

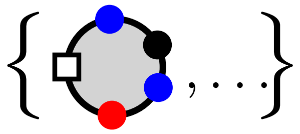
•, •, • are all matched.

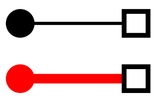
THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees



Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

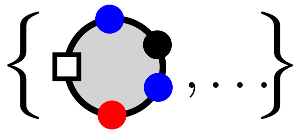
non-negative $\mathcal{Q}$-tree = necklace tree s.t.
necklaces are in $\mathcal{Q}$
the excess at each pearl is non negative.

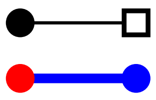
$\mathcal{Q} =$ 



•, • are all matched.
#{•} ≥ #{•}
in planted subtrees

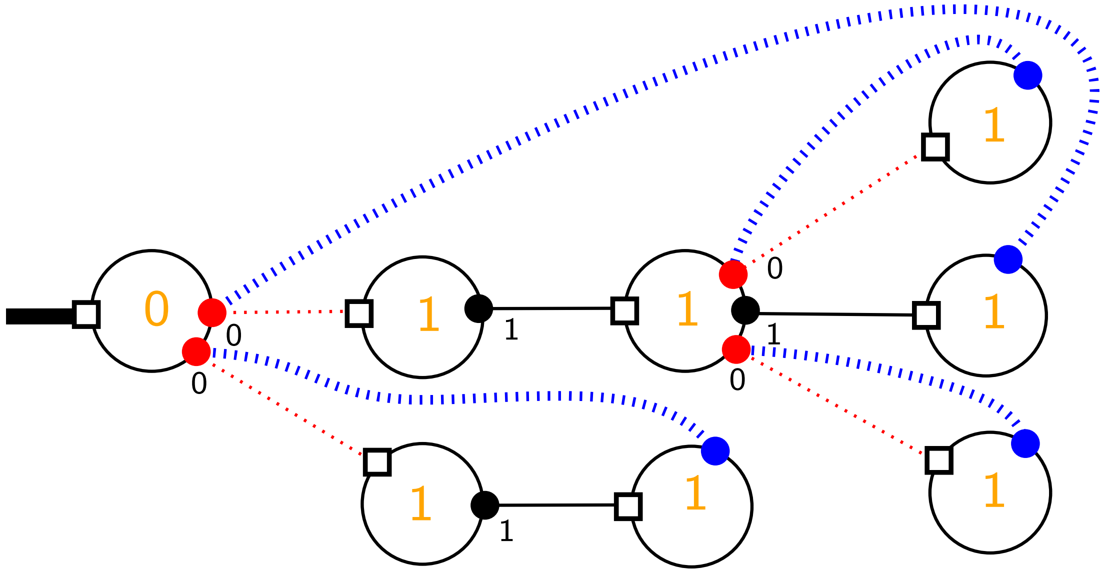
companion $\mathcal{Q}$- tree = necklace tree s.t.
necklaces are in $\mathcal{Q}$

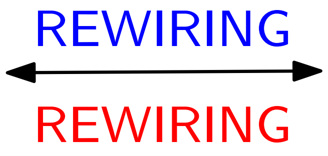
$\mathcal{Q} =$ 

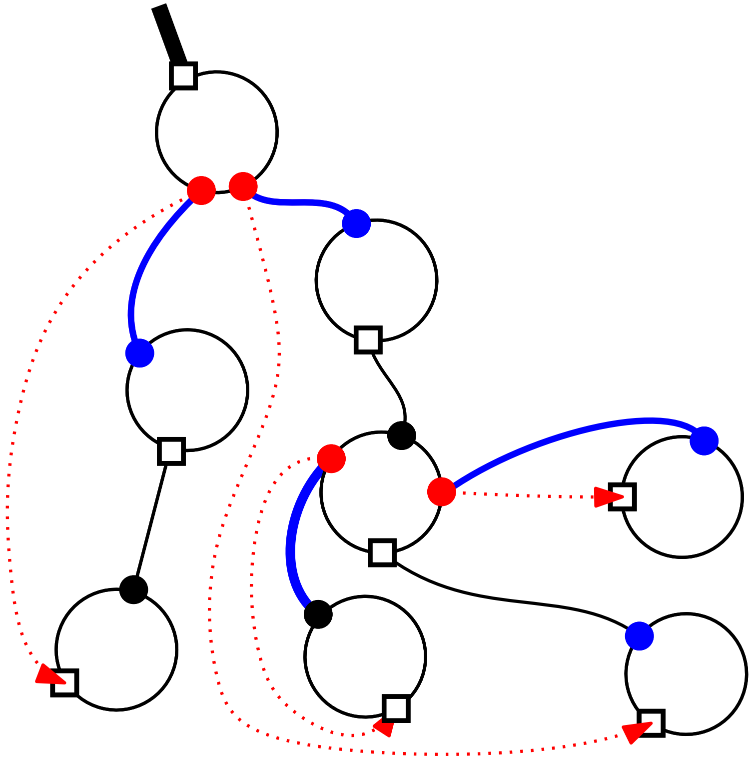


•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees





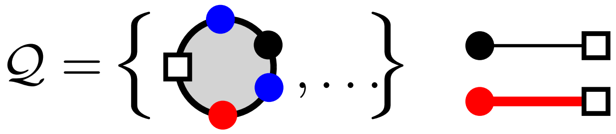


Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$

the excess at each pearl is non negative.

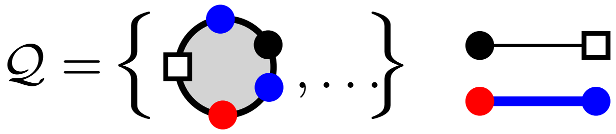


•, • are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$ in planted subtrees

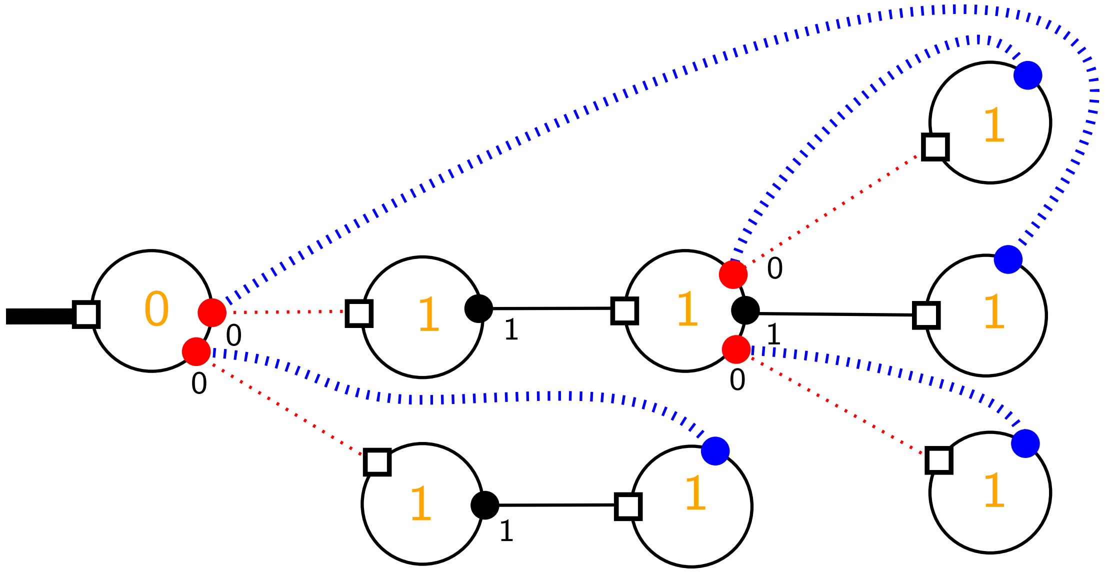
companion $\mathcal{Q}$- tree = necklace tree s.t.

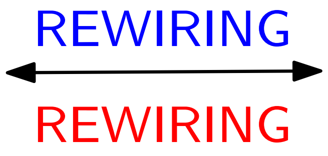
necklaces are in $\mathcal{Q}$

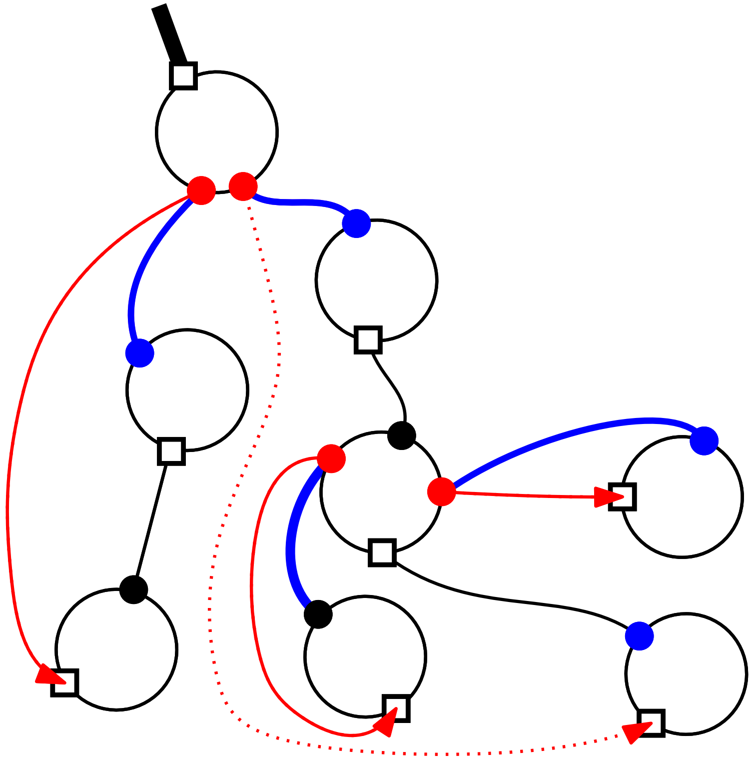


•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees





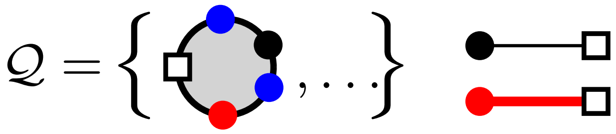


Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree $=$ necklace tree s.t.

necklaces are in $\mathcal{Q}$

the excess at each pearl is non negative.

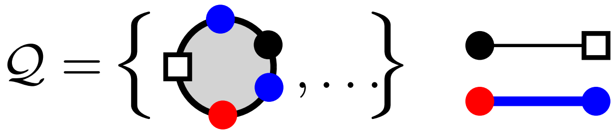


●, ● are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$
in planted subtrees

companion $\mathcal{Q}$- tree $=$ necklace tree s.t.

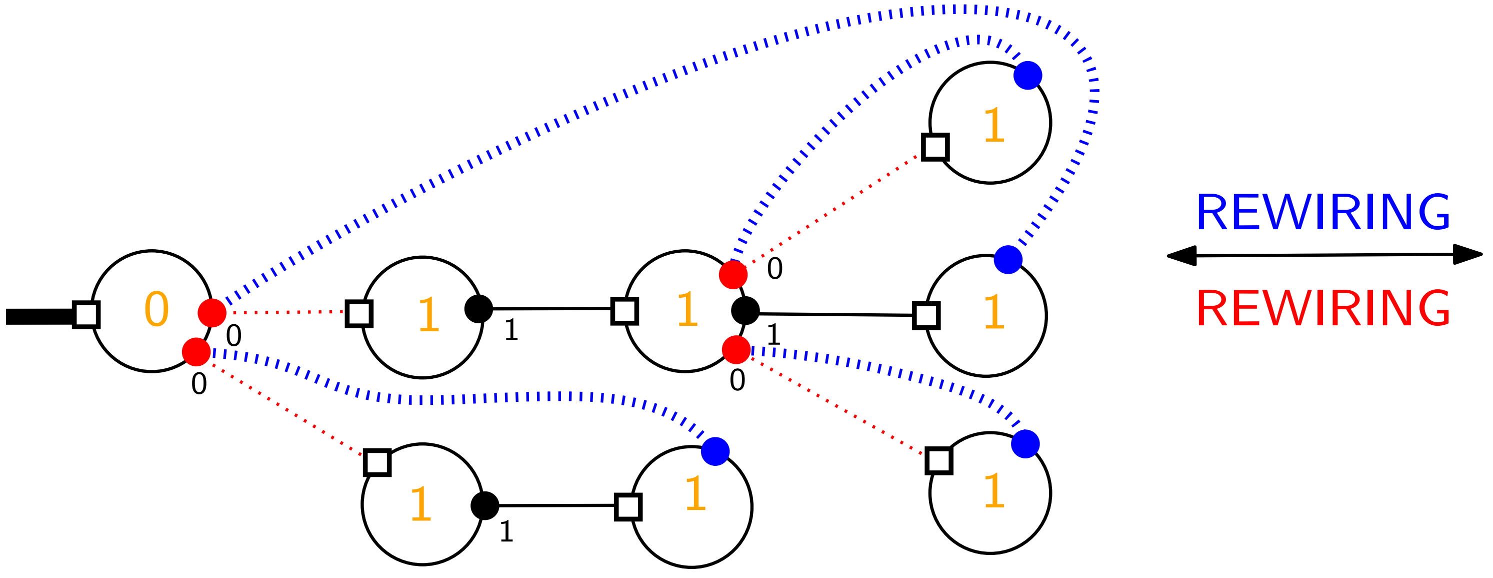
necklaces are in $\mathcal{Q}$

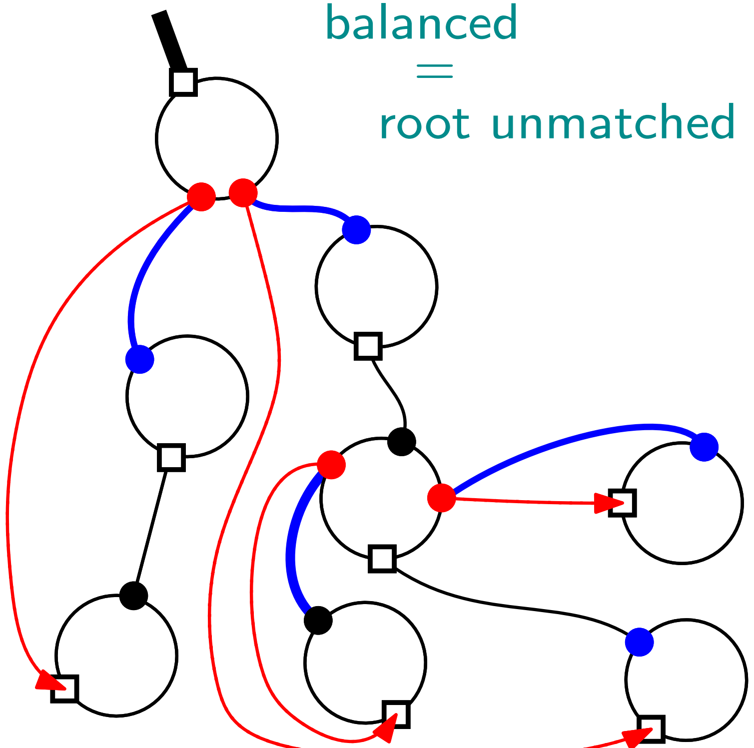


●, ●, ● are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees

balanced
=
root unmatched



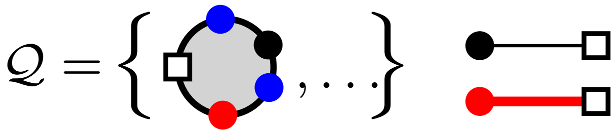


Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$

the excess at each pearl is non negative.



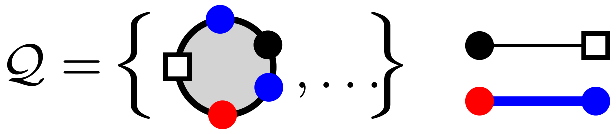
•, • are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$

in planted subtrees

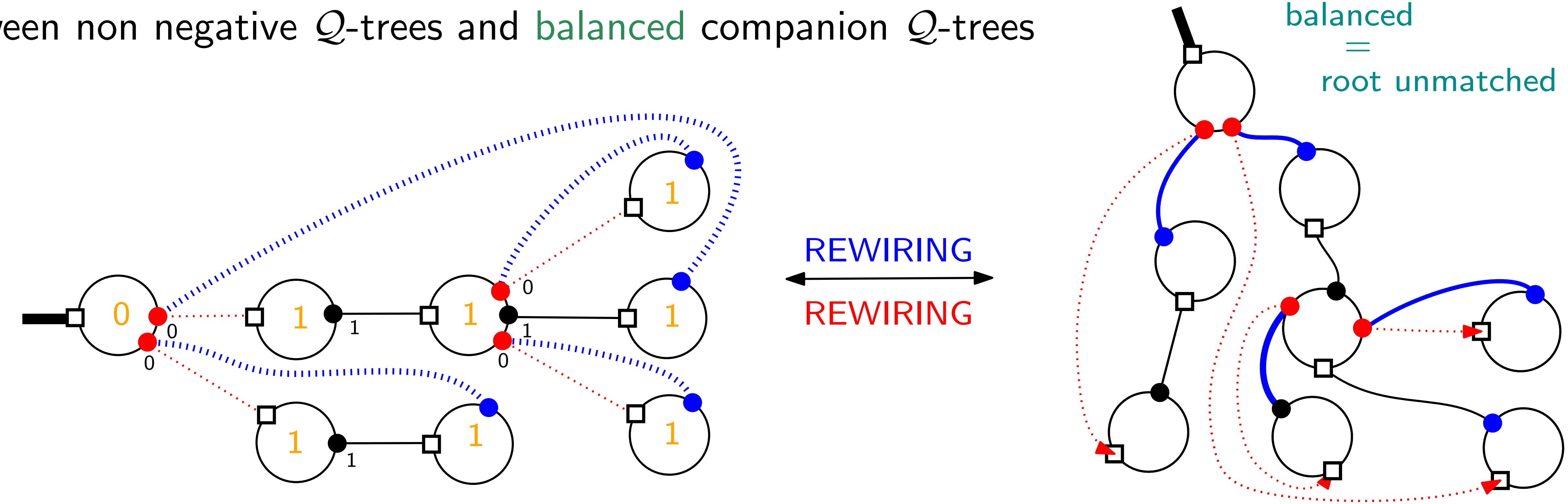
companion $\mathcal{Q}$- tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$

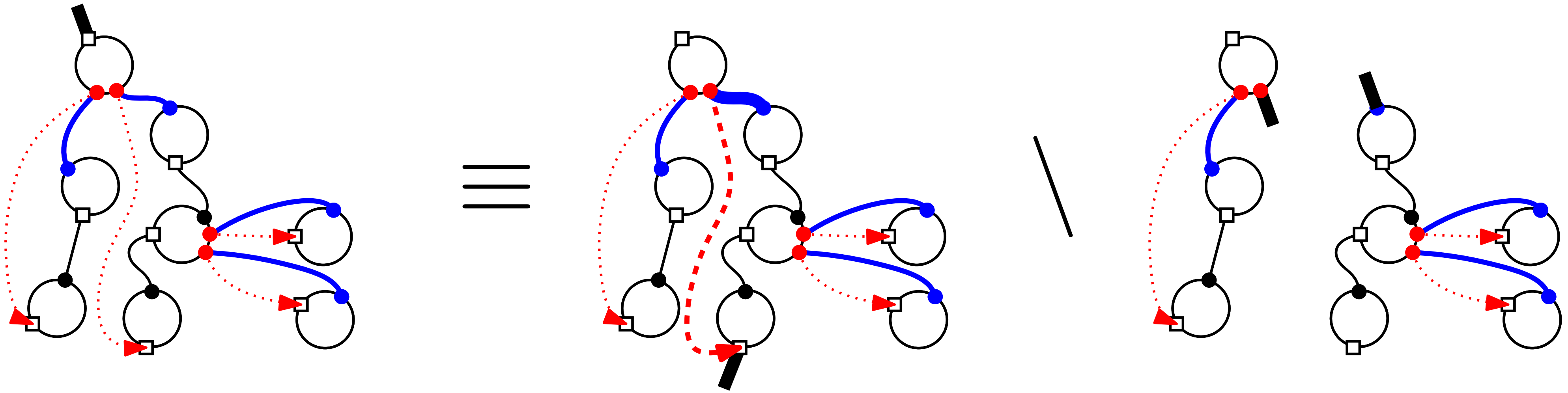


•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees

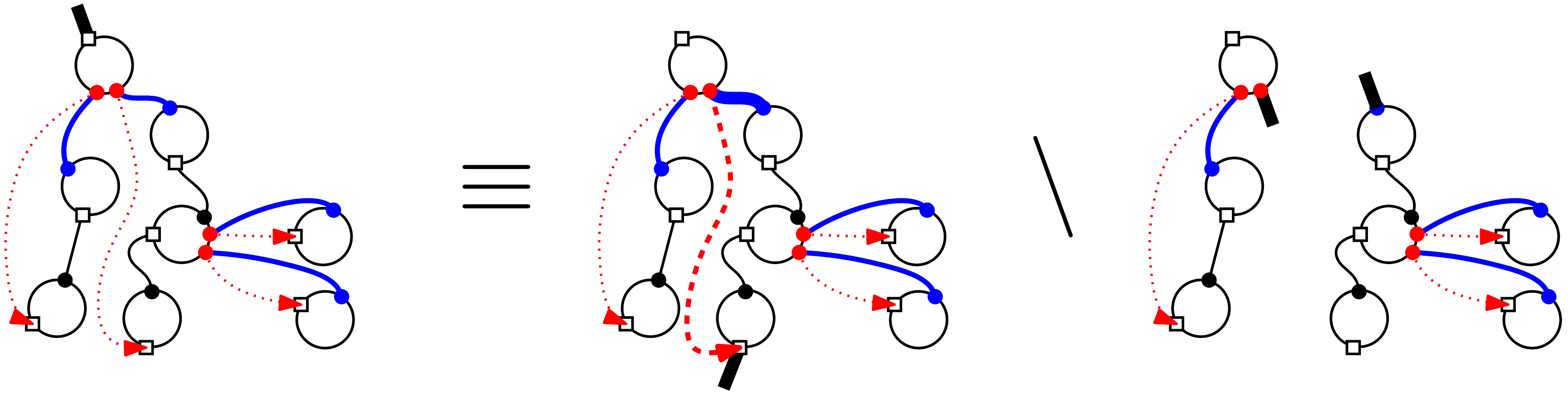


Balanced companion $\mathcal{Q}$-trees VS rooted companion $\mathcal{Q}$-trees

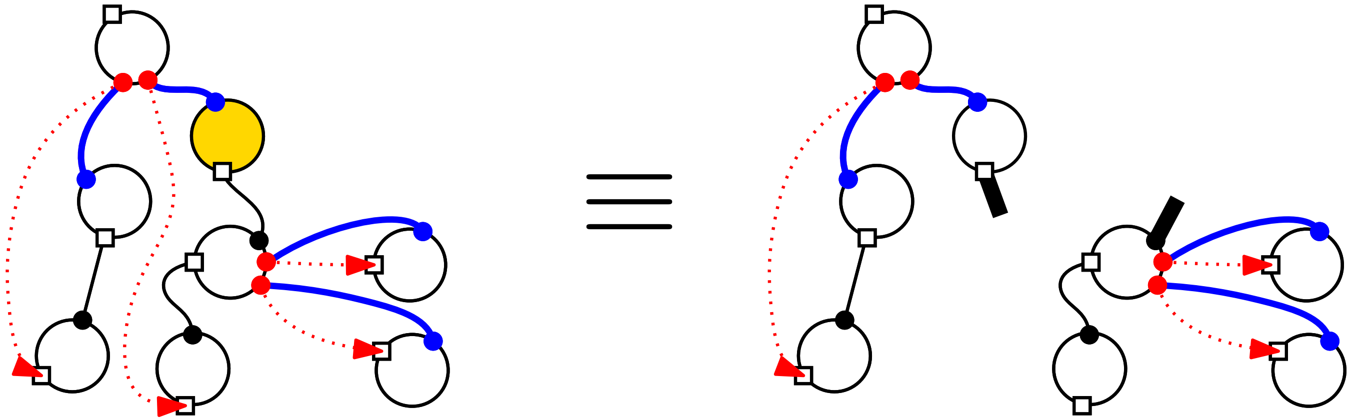


$$\mathcal{BC}_{\square} \equiv \mathcal{C}_{\square} - \mathcal{C}_{\color{red}\bullet} \times \mathcal{C}_{\color{blue}\bullet}$$

Balanced companion $\mathcal{Q}$-trees VS rooted companion $\mathcal{Q}$-trees



$$\mathcal{BC}_\square \equiv \mathcal{C}_\square - \mathcal{C}_{\color{red}\bullet} \times \mathcal{C}_{\color{blue}\bullet}$$

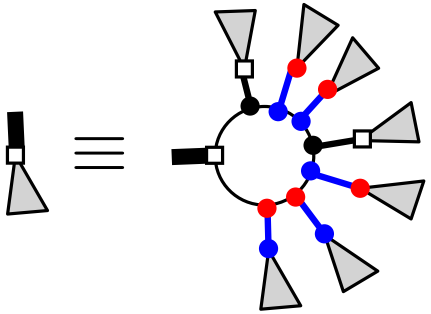


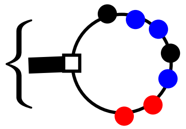
$$\mathcal{C}^{\circ} \equiv \mathcal{C}_\square \times (\varepsilon + \mathcal{C}_\bullet)$$

Context-free specifications for companion $\mathcal{Q}$-trees

$$C_{\square} = \mathcal{Z} \times Q(C_{\square}, C_{\textcolor{blue}{\bullet}}, C_{\textcolor{red}{\bullet}})$$

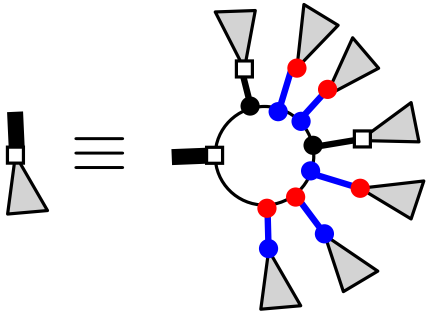
$$Q(v, w, u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{\textcolor{red}{\bullet}(s)} u^{\textcolor{blue}{\bullet}(s)}$$

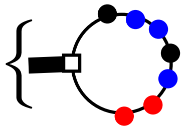


$\mathcal{Q} = \{$  $, \dots \}$

Context-free specifications for companion $\mathcal{Q}$-trees

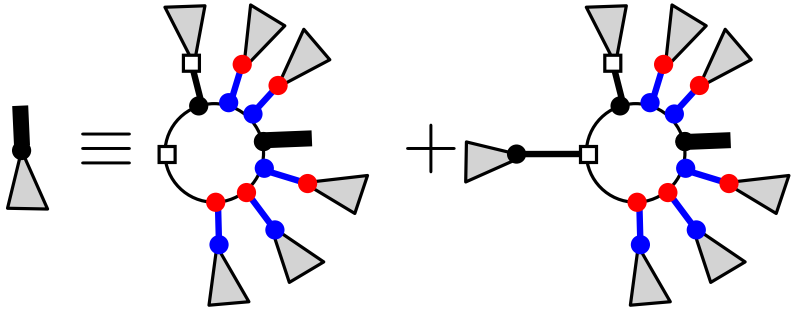
$$C_{\square} \;=\; \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

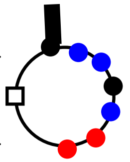
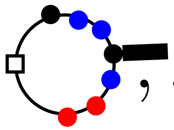


$\mathcal{Q} = \{$  $, \dots \}$

$$Q(v,w,u) = \sum_{s\in\mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$$

$$C_{\bullet} \;=\; \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

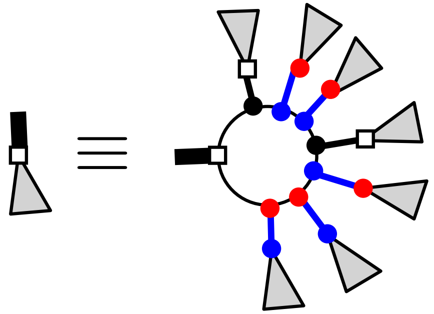


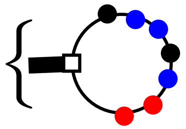
$Q'_{\bullet} = \{$  $,$  $, \dots \}$

Context-free specifications for companion $\mathcal{Q}$-trees

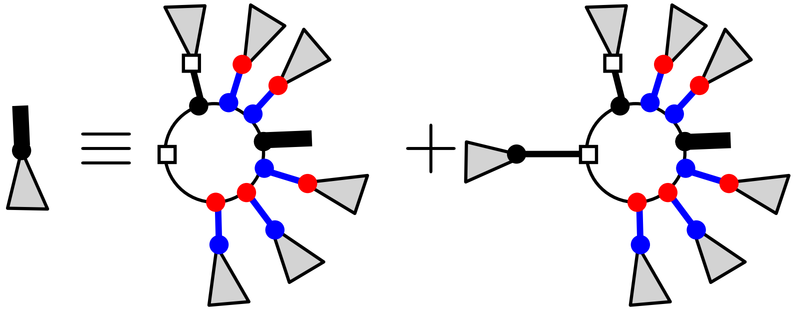
$$C_{\square} = \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

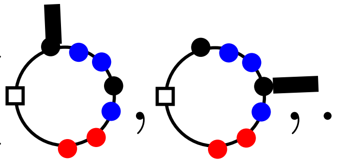
$$Q(v,w,u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$$



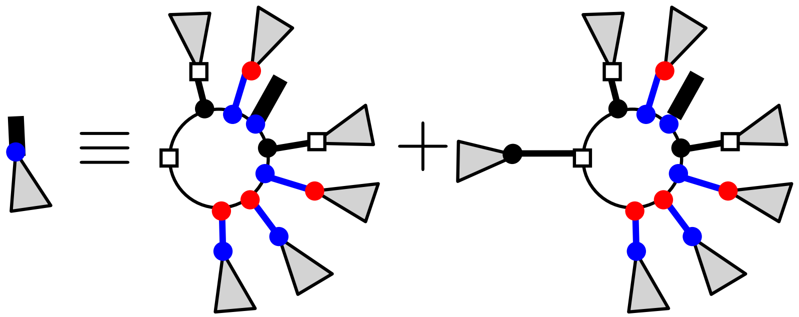
$\mathcal{Q} = \{$  $, \ldots \}$

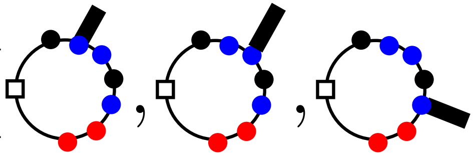
$$C_{\bullet} = \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



$Q'_{\bullet} = \{$  $, \ldots \}$

$$C_{\color{blue}\bullet} = \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\color{blue}\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



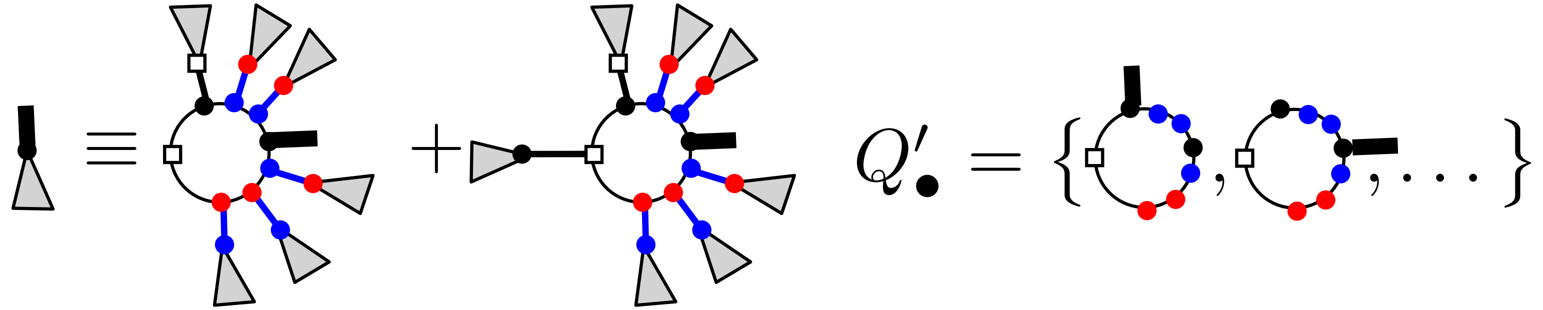
$Q'_{\color{blue}\bullet} = \{$  $\}$

Context-free specifications for companion $\mathcal{Q}$-trees

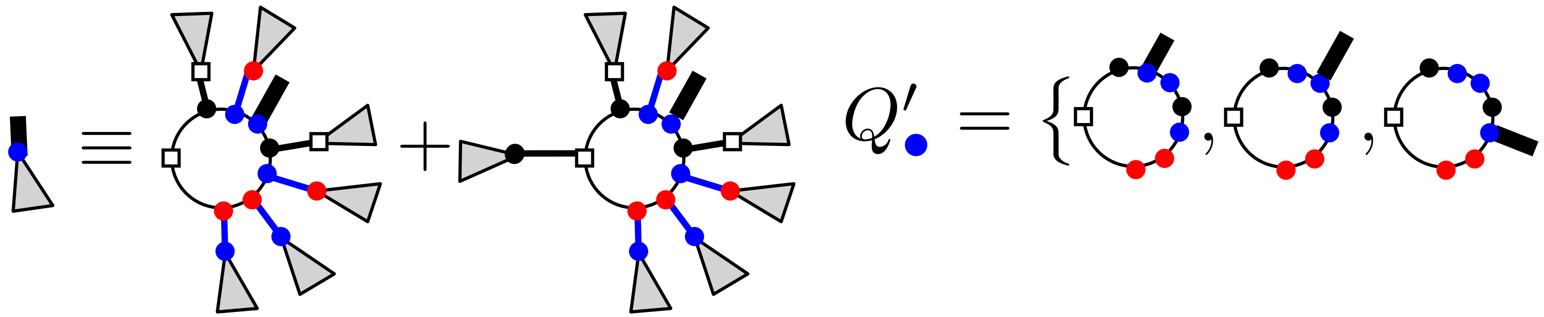
$$C_{\square} = \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

$$Q(v, w, u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$$

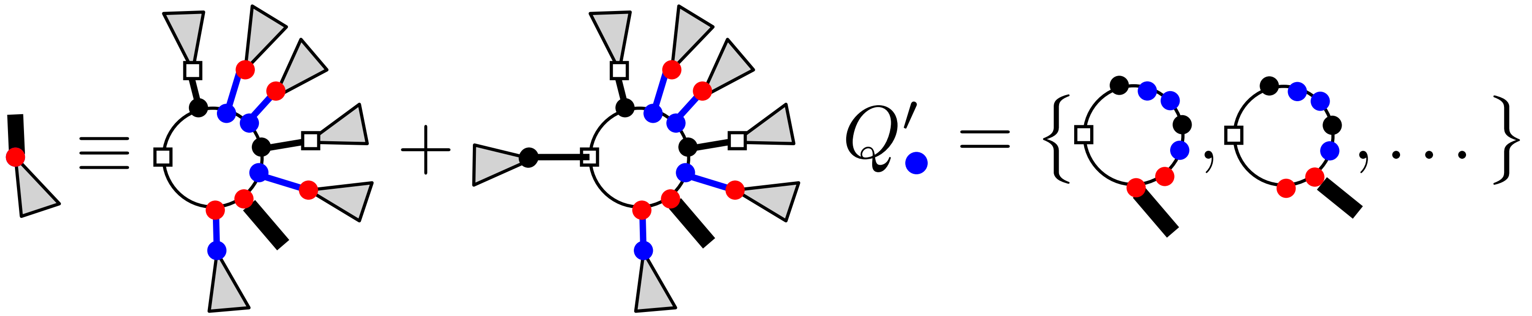
$$C_{\bullet} = \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



$$C_{\color{blue}\bullet} = \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\color{blue}\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



$$C_{\color{red}\bullet} = \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\color{red}\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



The combinatorial lifting of BMJ theorem

THEOREM (Duchi-S. 23)

Let $\mathcal{F} \equiv \mathcal{Z} \times Q\Big(\mathcal{F}, \frac{1}{u}(\mathcal{F} \setminus f), u\Big)$ be a catalytic decomposition of order one

where $Q(v,w,u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$ is the node gf of the associated non negative derivation $\mathcal{Q}$-trees

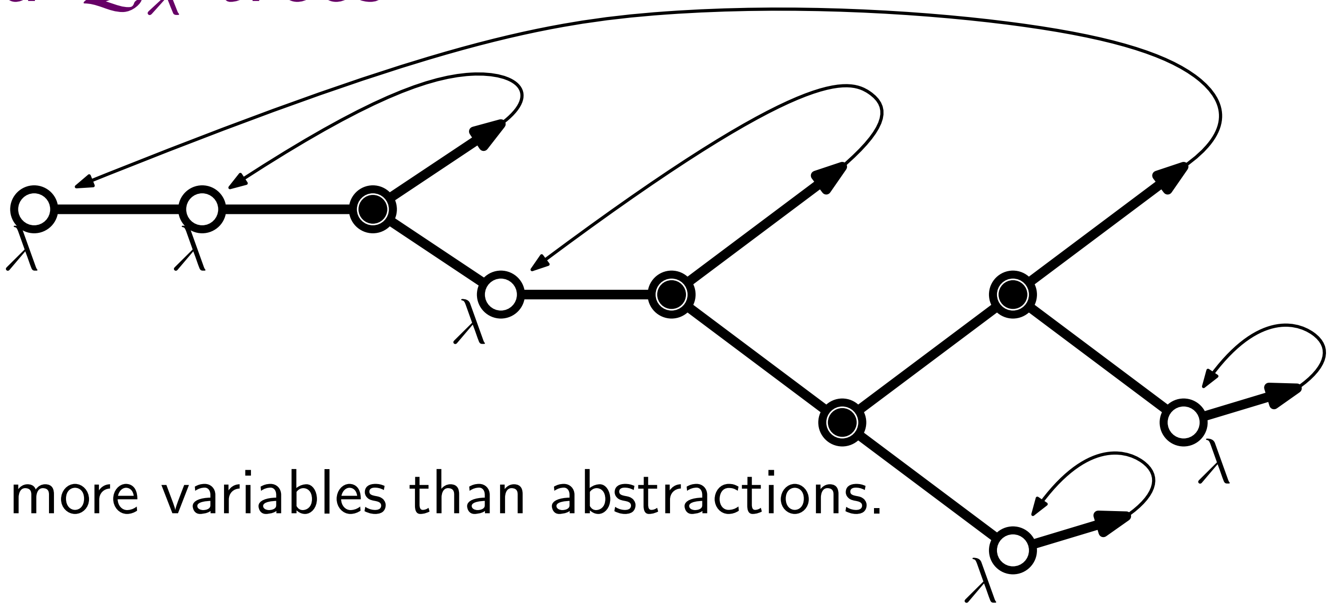
then

$$
\begin{array}{lclcr}
f & \overset{\text{rewiring}}{\equiv} & C & = & C_{\square} - C_{\color{blue}\bullet} \times C_{\color{red}\bullet} \\
f'_t & \overset{\text{rewiring}}{\equiv} & C^{\circ} & = & (1 + C_{\bullet}) \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})
\end{array}
$$

where the companion trees satisfy:

$$
\left\{
\begin{array}{lcl}
C_{\square} & = & \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet}) \\
C_{\bullet} & = & \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet}) \\
C_{\color{red}\bullet} & = & \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\color{red}\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet}) \\
C_{\color{blue}\bullet} & = & \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\color{blue}\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})
\end{array}
\right.
$$

Planar λ-terms and $\mathcal{Q}_\lambda$-trees



Open planar λ-term are to plane trees with

- applications: binary nodes
- abstractions: unary nodes
- variables: leaves, represented as arrow.

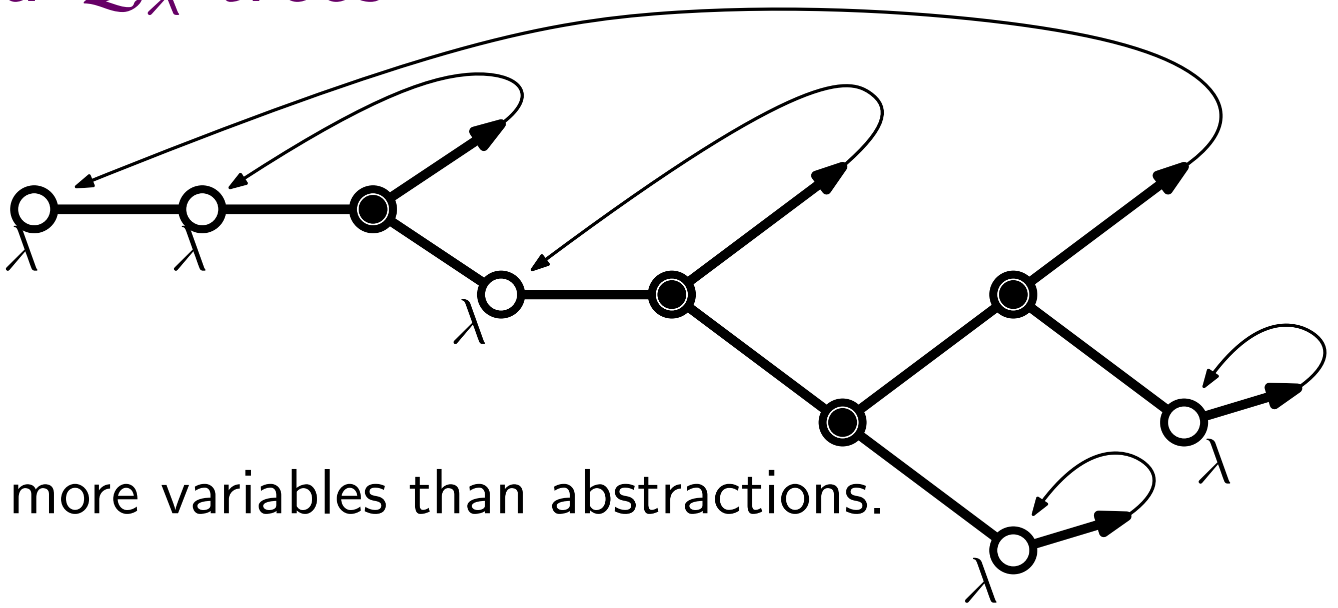
with condition that in each subterm there are more variables than abstractions.

Planar λ-terms and $\mathcal{Q}_\lambda$-trees

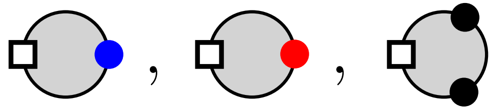
Open planar λ-term are to plane trees with

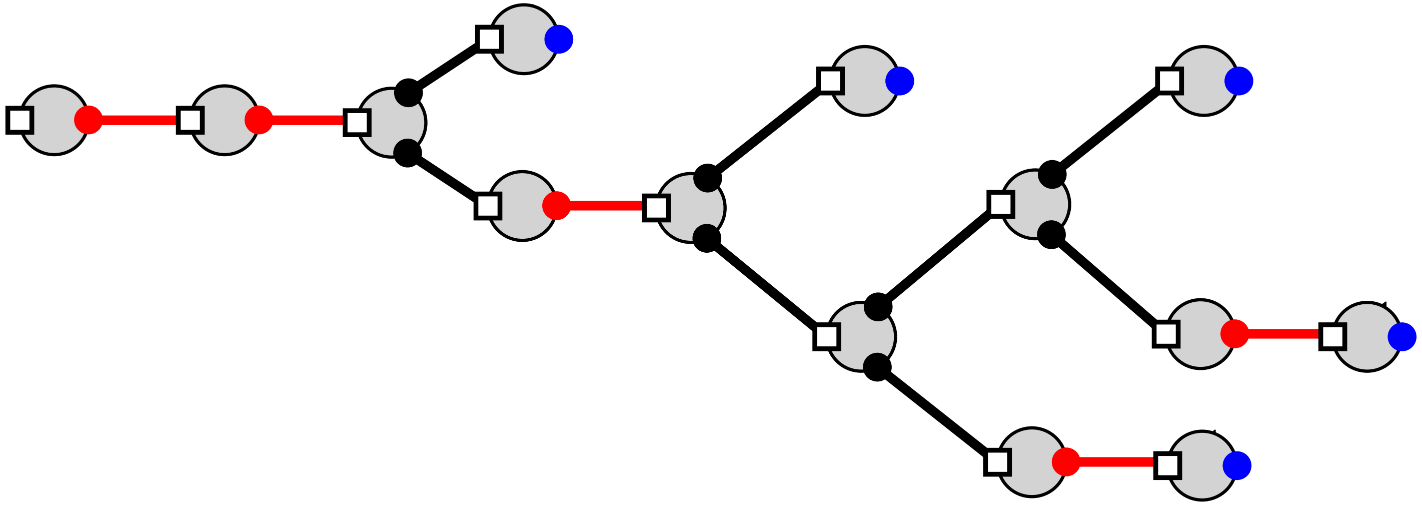
- applications: binary nodes
- abstractions: unary nodes
- variables: leaves, represented as arrow.

with condition that in each subterm there are more variables than abstractions.



Mark variables with $\color{blue}\bullet$ and abstractions λ with $\color{red}\bullet$,
then the set of vertex types is

$\mathcal{Q}_\lambda = \{$  $\}$



Then **non negative $\mathcal{Q}_\lambda$-trees =**
open planar λ-terms

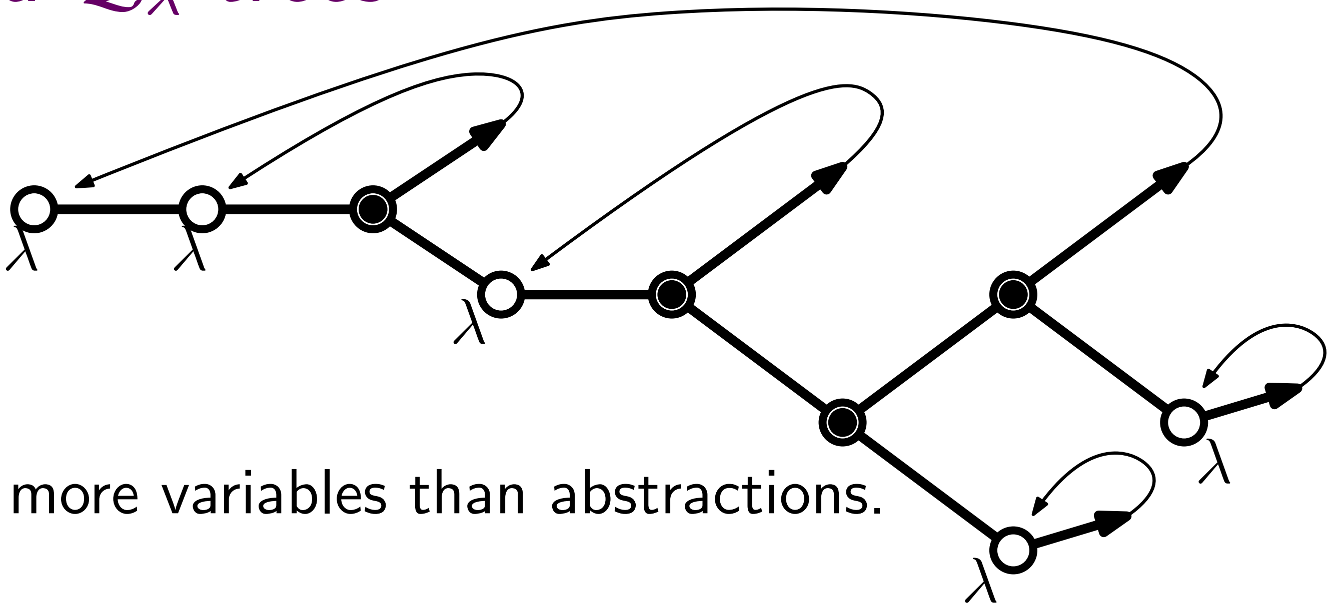
non negative $\mathcal{Q}_\lambda$-trees with excess 0 =
closed planar λ-terms

Planar λ-terms and $\mathcal{Q}_\lambda$-trees

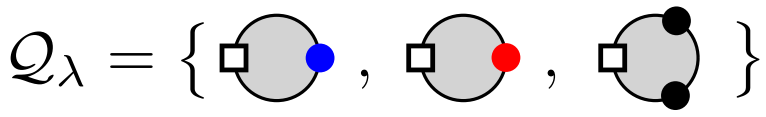
Open planar λ-term are to plane trees with

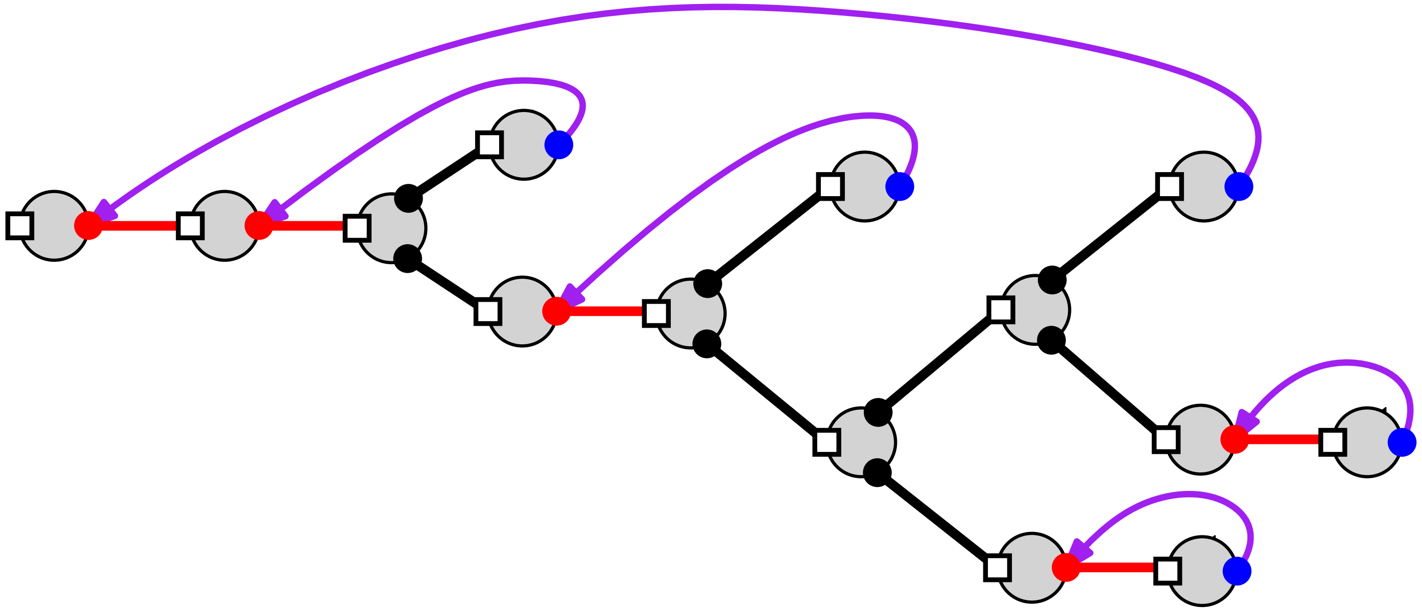
- applications: binary nodes
- abstractions: unary nodes
- variables: leaves, represented as arrow.

with condition that in each subterm there are more variables than abstractions.



Mark variables with • and abstractions λ with •,
then the set of vertex types is



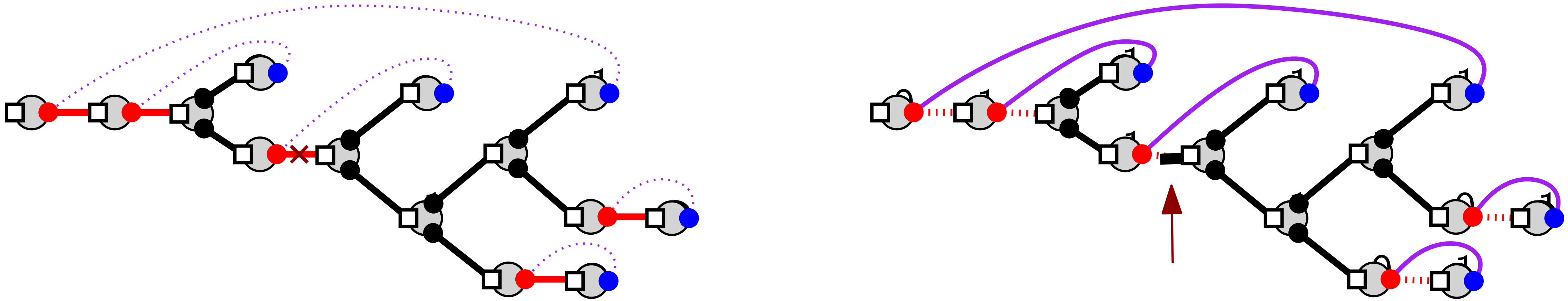


Then **non negative $\mathcal{Q}_\lambda$-trees = open planar λ-terms**

non negative $\mathcal{Q}_\lambda$-trees with excess 0 = closed planar λ-terms

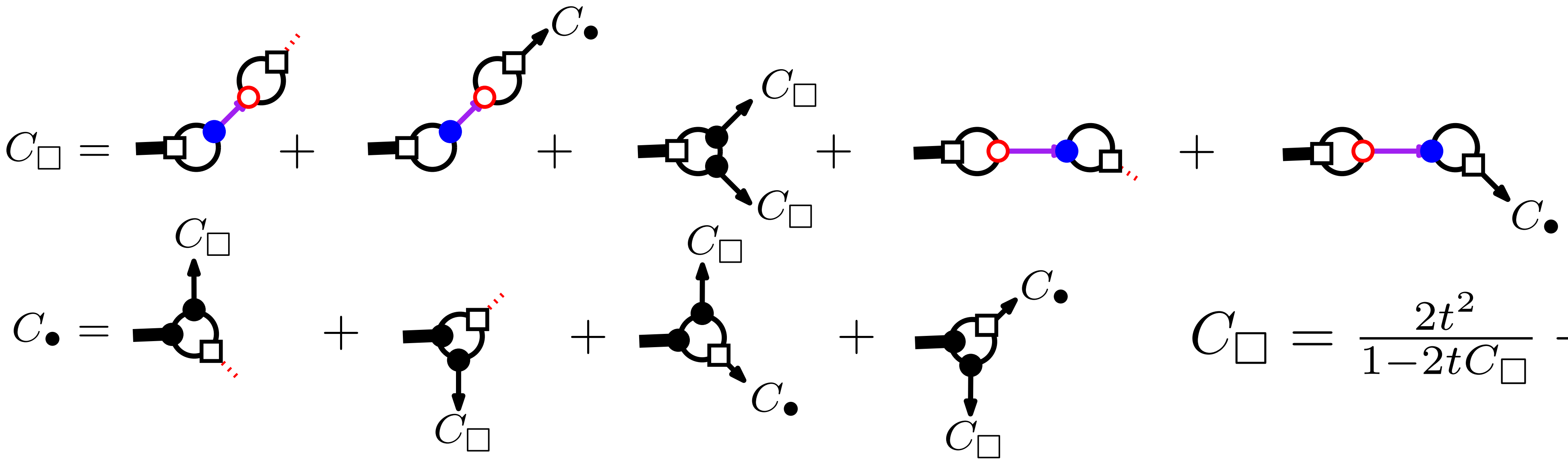
The closure corresponds to the rightmost depth first search abstraction-variable binding.

Planar λ-terms, closure and rewiring



Corollary.

Rewiring yields a size-preserving bijection between marked planar λ-terms and companion trees with context-free specification:



$$C_\square = \frac{2t^2}{1-2tC_\square} + tC_\square^2$$

What's next?

Catalytic equations also surface in various other enumeration problems, for instance for

- Families of pattern avoiding permutations (Zeilberger 92, Bona, Bousquet-Mélou, late 90's)
- Families of Tamari intervals (Chapoton, 2000's, Bousquet-Mélou-Chapoton 2022)
- Families of Planar (normal) λ-terms (Zeilberger and Giorgietti, 2015)
- Fighting fish and variants (Duchi et al, 2016)
- Fully parked trees (Chen 2021, Contat et al 2023)
- ...

Simply generated trees can be generated in linear time (Sportiello'21)

$\Rightarrow$ in principle yields linear time random generators for all these structures

Rewiring gives bijections between their catalytic derivation trees and simply generated multi-trees...

$\Rightarrow$ but can we also have direct context-free decompositions ? (cf *pizza slice* decompositions of maps)

Bijections allow to tackle new parameters...

$\Rightarrow$ so what is the equivalent of *distances in maps* for these structures ?

Thank you for you attention!

What's next? Higher order 1-catalytic equations

For order 1 we started from

$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u) - f), u\right),$ where $f \equiv f(t) = F(t, 0)$.

and the $\mathbb{N}$-algebraic system

$$\begin{cases} V &= t \cdot Q(V, W, U) \\ R &= t \cdot (1 + R) \cdot Q'_v(V, W, U) \\ U &= t \cdot (1 + R) \cdot Q'_w(V, W, U) \\ W &= t \cdot (1 + R) \cdot Q'_u(V, W, U) \end{cases}$$

For order k we need to deal with

$P\,(F(u), f_1, f_2, \ldots, f_k, u, t) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

What's next? Higher order 1-catalytic equations

For order 1 we started from

$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u) - f), u\right),$ where $f \equiv f(t) = F(t,0)$.

and the $\mathbb{N}$-algebraic system
$$\begin{cases} V &= t \cdot Q(V,W,U) \\ R &= t \cdot (1+R) \cdot Q'_v(V,W,U) \\ U &= t \cdot (1+R) \cdot Q'_w(V,W,U) \\ W &= t \cdot (1+R) \cdot Q'_u(V,W,U) \end{cases}$$

For order k we need to deal with

$P\left(F(u), f_1, f_2, \ldots, f_k, u, t\right) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \ldots, u_k$ of the series u, the $F(u_1), \ldots, F(u_k)$ by F and the $f_1, \ldots, f_k$.

What's next? Higher order 1-catalytic equations

For order 1 we started from

$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u) - f), u\right),$ where $f \equiv f(t) = F(t, 0)$.

and the $\mathbb{N}$-algebraic system
$$\begin{cases} V &=& t \cdot Q(V, W, U) \\ R &=& t \cdot (1+R) \cdot Q'_v(V, W, U) \\ U &=& t \cdot (1+R) \cdot Q'_w(V, W, U) \\ W &=& t \cdot (1+R) \cdot Q'_u(V, W, U) \end{cases}$$

For order k we need to deal with

$P\left(F(u), f_1, f_2, \ldots, f_k, u, t\right) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \ldots, u_k$ of the series u, the $F(u_1), \ldots, F(u_k)$ by F and the $f_1, \ldots, f_k$.

However this system is not immediately $\mathbb{N}$-algebraic

in fact the series u_i do not have non negative coefficients in general...

What's next? Higher order 1-catalytic equations

For order 1 we started from

$F(u) = t\,Q\left(F(u), \frac{1}{u}(F(u) - f), u\right),$ where $f \equiv f(t) = F(t,0)$.

and the $\mathbb{N}$-algebraic system
$$\begin{cases} V &= t \cdot Q(V,W,U) \\ R &= t \cdot (1+R) \cdot Q'_v(V,W,U) \\ U &= t \cdot (1+R) \cdot Q'_w(V,W,U) \\ W &= t \cdot (1+R) \cdot Q'_u(V,W,U) \end{cases}$$

For order k we need to deal with

$P\left(F(u), f_1, f_2, \ldots, f_k, u, t\right) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \ldots, u_k$ of the series u, the $F(u_1), \ldots, F(u_k)$ by F and the $f_1, \ldots, f_k$.

However this system is not immediately $\mathbb{N}$-algebraic

in fact the series u_i do not have non negative coefficients in general...

This is making things harder:

bijections are easier to find if one has a nice (**and** complicated) formula to interpret!

What's next? Higher order 1-catalytic equations

$P\,(F(u), f_1, f_2, \ldots, f_k, u, t) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \ldots, u_k$ of the series u, the $F(u_1), \ldots, F(u_k)$ by F and the $f_1, \ldots, f_k$.

What's next? Higher order 1-catalytic equations

$P\,(F(u), f_1, f_2, \ldots, f_k, u, t) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \ldots, u_k$ of the series u, the $F(u_1), \ldots, F(u_k)$ by F and the $f_1, \ldots, f_k$.

So here is the plan...

The linear case: essentially the kernel method for 1d walks with arbitray up and down steps

$\rightarrow$ the kernel method works systematically for finite sets of steps (Bousquet-Mélou, around 2000)

$\rightarrow$ the corresponding generalized Dyck path admit a context-free specification (Duchon 1998)

What's next? Higher order 1-catalytic equations

$P\left(F(u), f_1, f_2, \ldots, f_k, u, t\right) = 0$ or $P(u) = Q(F(u), \Delta F(u), \ldots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \ldots, u_k$ of the series u, the $F(u_1), \ldots, F(u_k)$ by F and the $f_1, \ldots, f_k$.

So here is the plan...

The linear case: essentially the kernel method for 1d walks with arbitray up and down steps

$\rightarrow$ the kernel method works systematically for finite sets of steps (Bousquet-Mélou, around 2000)

$\rightarrow$ the corresponding generalized Dyck path admit a context-free specification (Duchon 1998)

New observation: Rewriting the equations satisfied by the kernel roots u_i in terms of the elementary symmetric functions in the 'finite' and 'infinite' root separately directy yields Duchon's $\mathbb{N}$-algebraic equations.

$\Rightarrow$ gives a combinatorial specifications for walks with algebraic series of up-steps.

What's next? Higher order 1-catalytic equations

$P\left(F(u), f_1, f_2, \dots, f_k, u, t\right) = 0$ or $P(u) = Q(F(u), \Delta F(u), \dots \Delta^k F(u), u, t)$

BMJ Theorem for order k equations lead to a system of $3k$ equations for $3k$ unknowns: the analogs $u_1, \dots, u_k$ of the series u, the $F(u_1), \dots, F(u_k)$ by F and the $f_1, \dots, f_k$.

So here is the plan...

The linear case: essentially the kernel method for 1d walks with arbitray up and down steps

$\rightarrow$ the kernel method works systematically for finite sets of steps (Bousquet-Mélou, around 2000)

$\rightarrow$ the corresponding generalized Dyck path admit a context-free specification (Duchon 1998)

New observation: Rewriting the equations satisfied by the kernel roots u_i in terms of the elementary symmetric functions in the 'finite' and 'infinite' root separately directy yields Duchon's $\mathbb{N}$-algebraic equations.

$\Rightarrow$ gives a combinatorial specifications for walks with algebraic series of up-steps.

The non linear case: the resulting heuristic is to rewrite the BMJ systems in terms of the elementary functions in the u_i instead, and to avoid the $F(u_i)$, use the discriminant form of the sytem.

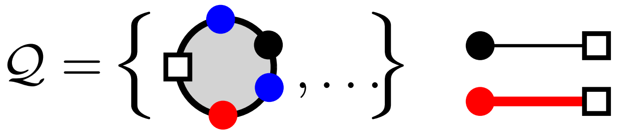
in progress: apply the combinatorial specification of the linear case along a branch and sort out the ugly details to see what comes out !

Non negative $\mathcal{Q}$-trees and companion $\mathcal{Q}$-trees

non-negative $\mathcal{Q}$-tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$

the excess at each pearl is non negative.



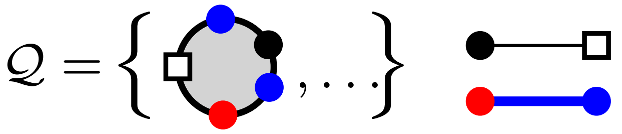
•, • are all matched.

$\#\{\bullet\} \geq \#\{\bullet\}$

in planted subtrees

companion $\mathcal{Q}$- tree = necklace tree s.t.

necklaces are in $\mathcal{Q}$



•, •, • are all matched.

THEOREM (Duchi-S. 23). **Rewiring** is a vertex type preserving bijection between non negative $\mathcal{Q}$-trees and balanced companion $\mathcal{Q}$-trees