On the combinatorics of one variable catalytic equations

GILLES SCHAEFFER
LIX, CNRS, Institut Polytechnique de Paris

join work with ENRICA DUCHI

L'esprit des cartes, une conférence
en l'honneur d'Emmanuel Guitter

Mai 15, 2025, Saclay

1-catalytic equations

Recall from Mireille's talk that a *1-catalytic equation* is an equation of the form

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

where P 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 $f_i \equiv f_i(t) \in \mathbb{F}[[t]]$.

1-catalytic equations

Recall from Mireille's talk that a *1-catalytic equation* is an equation of the form

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

where P 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 $f_i \equiv f_i(t) \in \mathbb{F}[[t]]$.

A nice example is the (Bender-Canfield generalized) *Tutte equation*

$$F(u) = 1 + tu^2 F(u)^2 + t \sum_{i \geq 2} z_i \frac{F(u) - \sum_{j=0}^{i-2} u^j F_j}{u^{i-2}}$$

for the enumeration of rooted planar maps w.r.t. face degree distribution.

1-catalytic equations

Recall from Mireille's talk that a *1-catalytic equation* is an equation of the form

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

where P 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 $f_i \equiv f_i(t) \in \mathbb{F}[[t]]$.

A nice example is the (Bender-Canfield generalized) *Tutte equation*

$$F(u) = 1 + tu^2 F(u)^2 + t \sum_{i \geq 2} z_i \frac{F(u) - \sum_{j=0}^{i-2} u^j F_j}{u^{i-2}}$$

for the enumeration of rooted planar maps w.r.t. face degree distribution.

More generally 1-catalytic equations are ubiquitous in map enumeration, and closely related to the *loop equations* of the early matrix integral literature. They are also sometimes known as *discrete differential equations*.

1-catalytic equations

Recall from Mireille's talk that a *1-catalytic equation* is an equation of the form

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

where P 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-catalytic equations have algebraic solutions

Recall from Mireille's talk that a *1-catalytic equation* is an equation of the form

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

where P 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-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.

1-catalytic equations have algebraic solutions

For the earlier Tutte equation

$$F(u) = 1 + tu^2F(u)^2 + tuz_1F(u) + \sum_{i\geq 2} z_i\Delta^{i-2}F(u)$$

BMJ theorem yields a parametrization that can be then rewritten as

$$F_2(t) = S_1^2 + S_2 - 2S_1[v^{-2}]W - [v^{-3}]W$$

with

$$W = \sum_{i\geq 1} z_i(v + S_1 + S_2/v)^{i-1}, \quad S_1 = t[u^0]W, \quad \text{and} \quad S_2 = t + t[v^{-1}]W.$$

1-catalytic equations have algebraic solutions

With $z_i = 0$ for all $i > m$.

For the earlier Tutte equation

$$F(u) = 1 + tu^2F(u)^2 + tuz_1F(u) + \sum_{i\geq 2} z_i\Delta^{i-2}F(u)$$

BMJ theorem yields a parametrization that can be then rewritten as

$$F_2(t) = S_1^2 + S_2 - 2S_1[v^{-2}]W - [v^{-3}]W$$

with

$$W = \sum_{i\geq 1} z_i(v + S_1 + S_2/v)^{i-1}, \quad S_1 = t[u^0]W, \quad \text{and} \quad S_2 = t + t[v^{-1}]W.$$

1-catalytic equations have algebraic solutions

With $z_i = 0$ for all $i > m$.

For the earlier Tutte equation

$$F(u) = 1 + tu^2 F(u)^2 + tuz_1 F(u) + \sum_{i\geq 2} z_i \Delta^{i-2} F(u)$$

BMJ theorem yields a parametrization that can be then rewritten as

$$F_2(t) = S_1^2 + S_2 - 2S_1[v^{-2}]W - [v^{-3}]W$$

with

$$W = \sum_{i\geq 1} z_i (v + S_1 + S_2/v)^{i-1}, \quad S_1 = t[u^0]W, \quad \text{and} \quad S_2 = t + t[v^{-1}]W.$$

For instance for triangulations, $z_i = 0$ for all $i \neq 3$, and we get:

$$F_2(t) = S_1^2 + S_2 - 2S_1 S_2^2$$

with

$$S_1 = t(S_1^2 + 2S_2), \quad \text{and} \quad S_2 = t + 2t(S_1 S_2).$$

Context free languages and algebraic specifications/decompositions

Algebraic equations are closely related to 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

Algebraic equations are closely related to 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. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

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

Context free languages and algebraic specifications/decompositions

Algebraic equations are closely related to 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. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

The 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)

Context free languages and algebraic specifications/decompositions

Algebraic equations are closely related to 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. $\mathcal{A} \equiv \mathbf{z} + \mathcal{A} \times \mathcal{A}$

The 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)

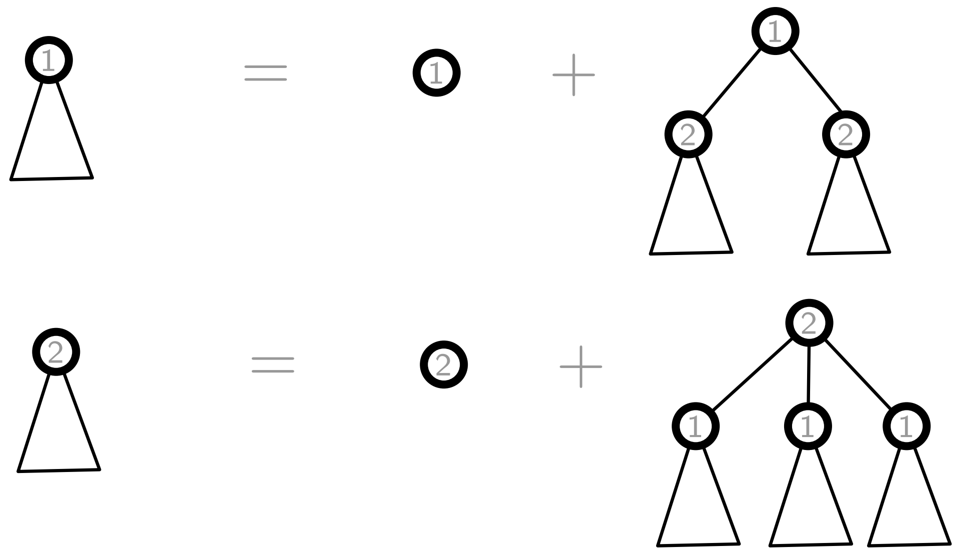
Conversely when the gf of a combinatorial familly $\mathcal{A}$ is known to be $\mathbb{N}$-algebraic, one would like to explain it via a **context-free specification** of $\mathcal{A}$.

(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:

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

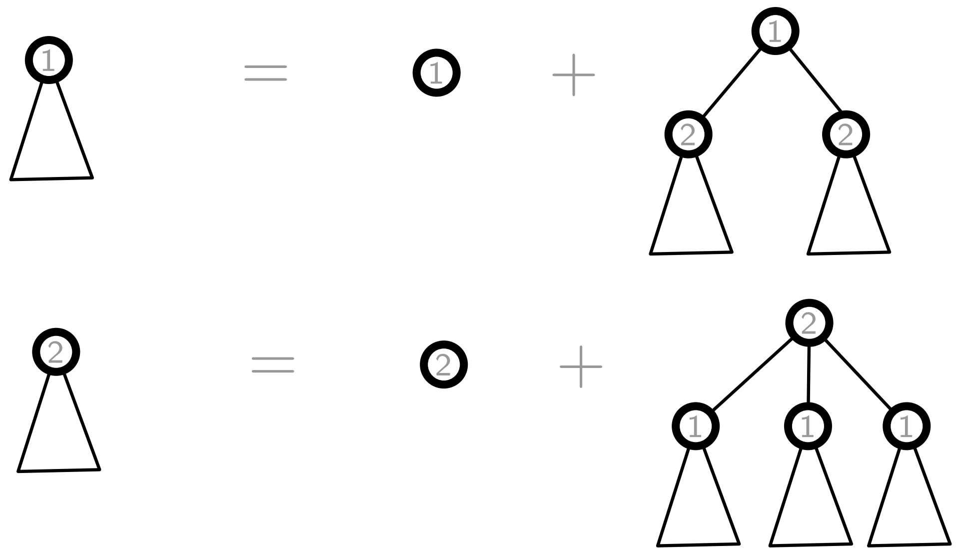


The *derivation trees of a context-free specification* are multitype simply generated trees, *i.e.* trees specified by the allowed node progeny for each color, with independent subtrees.

Context-free specifications and multitype simply generated trees

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

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



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**.

Context-free specification for maps

For the earlier Tutte equation

$$F(u) = 1 + tu^2 F(u)^2 + tuz_1 F(u) + \sum_{i\geq 2} z_i \Delta^{i-2} F(u)$$

BMJ theorem yields a parametrization that can be then rewritten as

$$F_2(t) = S_1^2 + S_2 - 2S_1[v^{-2}]W - [v^{-3}]W$$

with

$$W = \sum_{i\geq 1} z_i (v + S_1 + S_2/v)^{i-1}, \quad S_1 = t[u^0]W, \quad \text{and} \quad S_2 = t + t[v^{-1}]W.$$

This algebraic parametrization was given **two** beautiful combinatorial interpretations by **Emmanuel Guitter** with Jérémie Bouttier and Philippe Di Francesco

- first in terms of *blossoming trees*,

[Census of Planar Maps: From the One-Matrix Model Solution to a Combinatorial Proof, *Nuclear physics*, 2002]

- and then in terms of their celebrated *BDFG mobiles*.

[Planar maps as labeled mobiles, *Electr. J. Comb*, 2004]

Context-free specification for maps

For the earlier Tutte equation

$$F(u) = 1 + tu^2F(u)^2 + tuz_1F(u) + \sum_{i\geq 2} z_i\Delta^{i-2}F(u)$$

BMJ theorem yields a parametrization that can be then rewritten as

$$F_2(t) = S_1^2 + S_2 - 2S_1[v^{-2}]W - [v^{-3}]W$$

with

$$W = \sum_{i\geq 1} z_i(v + S_1 + S_2/v)^{i-1}, \quad S_1 = t[u^0]W, \quad \text{and} \quad S_2 = t + t[v^{-1}]W.$$

This algebraic parametrization was given **two** beautiful combinatorial interpretations by **Emmanuel Guitter** with Jérémie Bouttier and Philippe Di Francesco

- first in terms of *blossoming trees*,

[Census of Planar Maps: From the One-Matrix Model Solution to a Combinatorial Proof, *Nuclear physics*, 2002]

- and then in terms of their celebrated *BDFG mobiles*.

[Planar maps as labeled mobiles, *Electr. J. Comb*, 2004]

Later Emmanuel and Jérémie also gave a direct *context-free specification* of maps

Planar maps as pizza slices, *aka* [Planar Maps and continous fractions, *Comm. Math. Phys.*, 2012]

Catalytic specifications as a source of bijective problems

In summary:

- Maps admit "easy" catalytic specifications
- Catalytic equations have *nice* algebraic solutions

$\Rightarrow$ combinatorial interpretation problem !

These ideas have been generalized for a huge variety of map families...

and have led to many developments in Combinatorics, Probability or Algorithmics

Catalytic specifications as a source of bijective problems

In summary:

- Maps admit "easy" catalytic specifications
- Catalytic equations have *nice* algebraic solutions

$\Rightarrow$ combinatorial interpretation problem !

These ideas have been generalized for a huge variety of map families...

and have led to many developments in Combinatorics, Probability or Algorithmics

But 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)
- ...

Catalytic specifications as a source of bijective problems

In summary:

- Maps admit "easy" catalytic specifications
- Catalytic equations have *nice* algebraic solutions

$\Rightarrow$ combinatorial interpretation problem !

These ideas have been generalized for a huge variety of map families...

and have led to many developments in Combinatorics, Probability or Algorithmics

But

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)
- ...

So could we (should we?) carry on this combinatorial interpretation program for all these objects...

Catalytic specifications as a source of bijective problems

In summary:

- Maps admit "easy" catalytic specifications
- Catalytic equations have *nice* algebraic solutions

$\Rightarrow$ combinatorial interpretation problem !

These ideas have been generalized for a huge variety of map families...

and have led to many developments in Combinatorics, Probability or Algorithmics

But 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)
- ...

So could we (should we?) carry on this combinatorial interpretation program for all these objects...

What about a systematic approach?

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

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

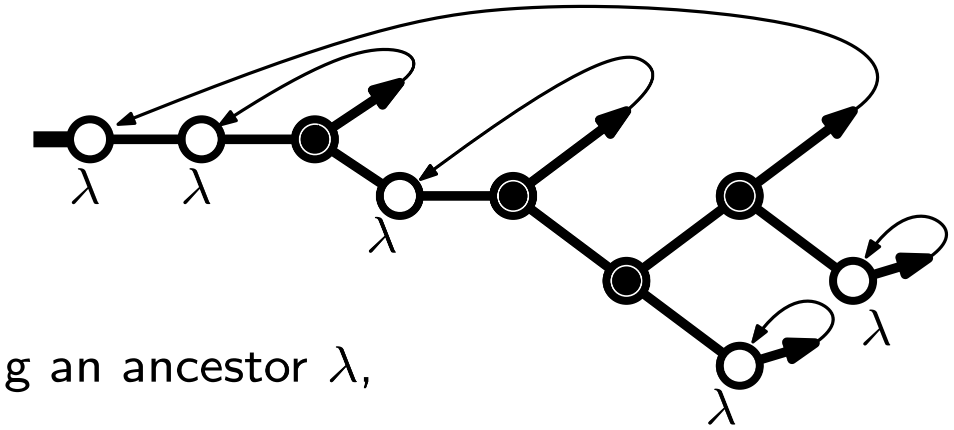
Let U, V, W and R be the unique fps satisfying the system

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

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

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

The example of planar λ-terms

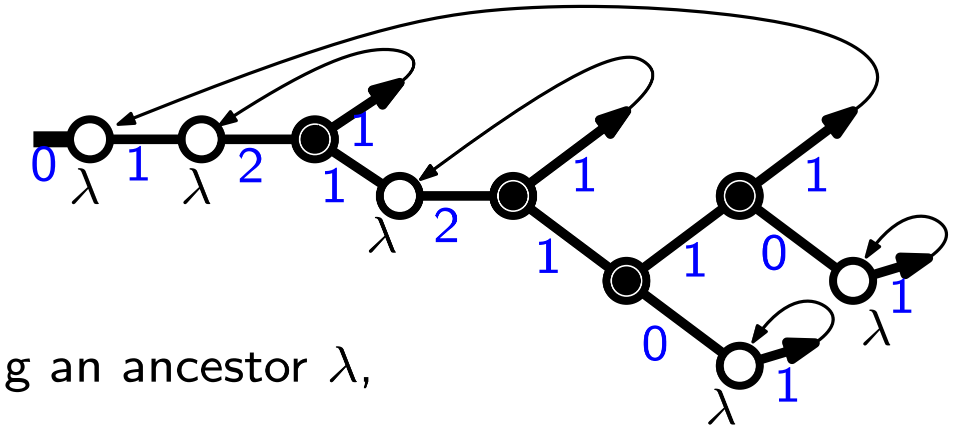


Planar λ-terms can be presented as trees with

- **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

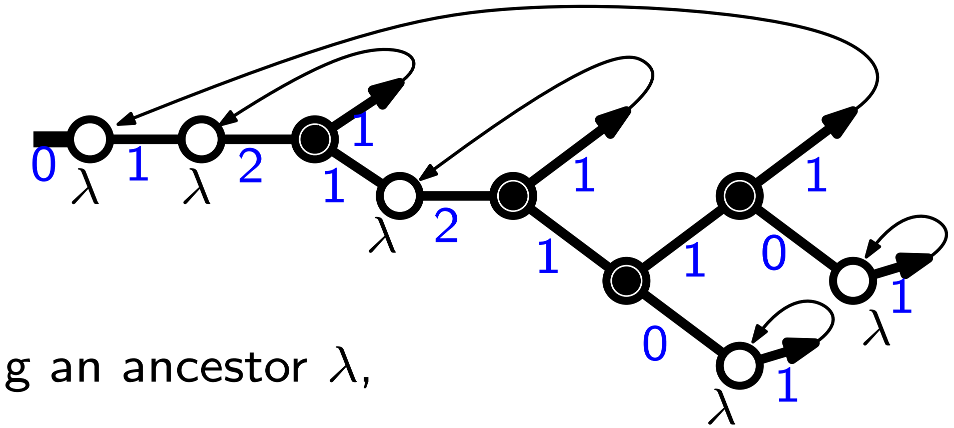
- **applications**: binary nodes
- **λ-abstractions**: unary nodes
- **variables**: leaves, represented as arrows , each matching an ancestor λ,

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

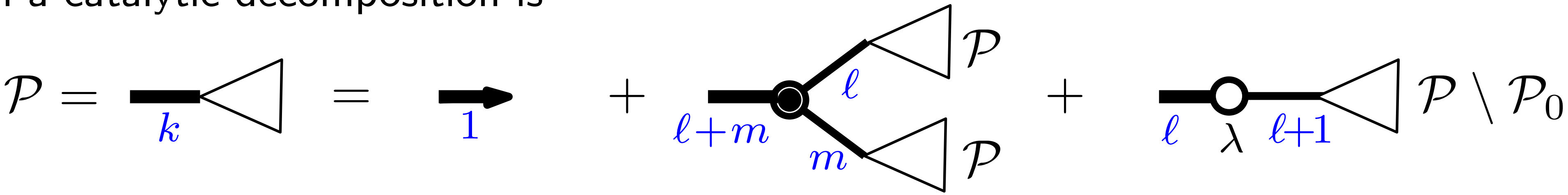
- **applications**: binary nodes
- **λ-abstractions**: unary nodes
- **variables**: leaves, represented as arrows, each matching an ancestor λ,

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

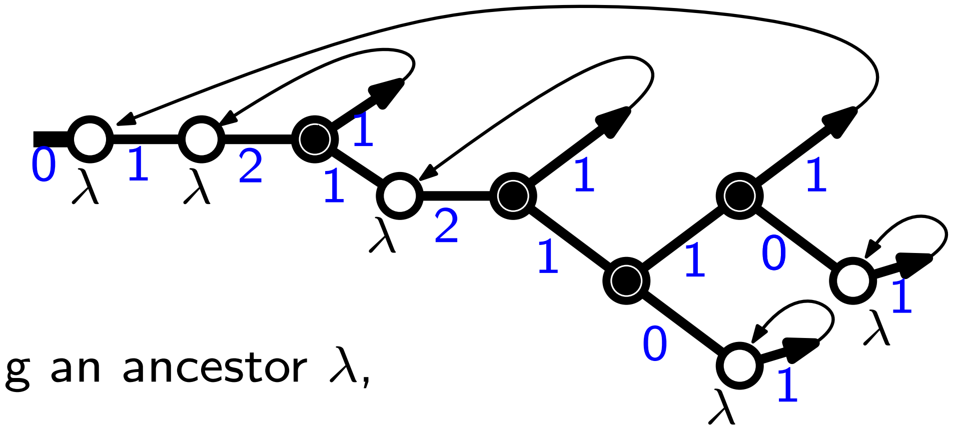
Equivalently, in each subterm there are more variables than abstractions,

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

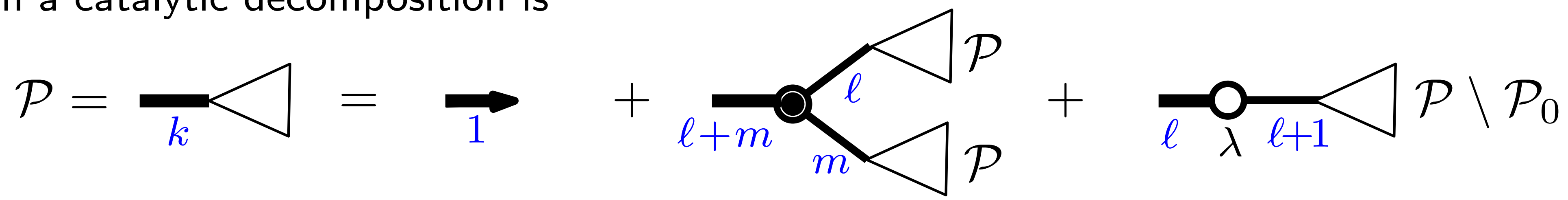
- **applications**: binary nodes
- **λ-abstractions**: unary nodes
- **variables**: leaves, represented as arrows , each matching an ancestor λ,

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



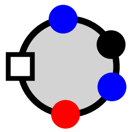
and the catalytic equation for the gf $P(u) = \sum_{\tau \in \mathcal{P}} t^{|\tau|} u^{excess(\tau)}$ is

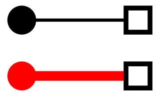
$$P(u) \quad = \quad tu \quad + \quad tP(u)^2 \quad + \quad \frac{t}{u}(P(u) - P(0))$$

Decorated trees and non negative trees

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

the excess at each pearl is non negative.

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



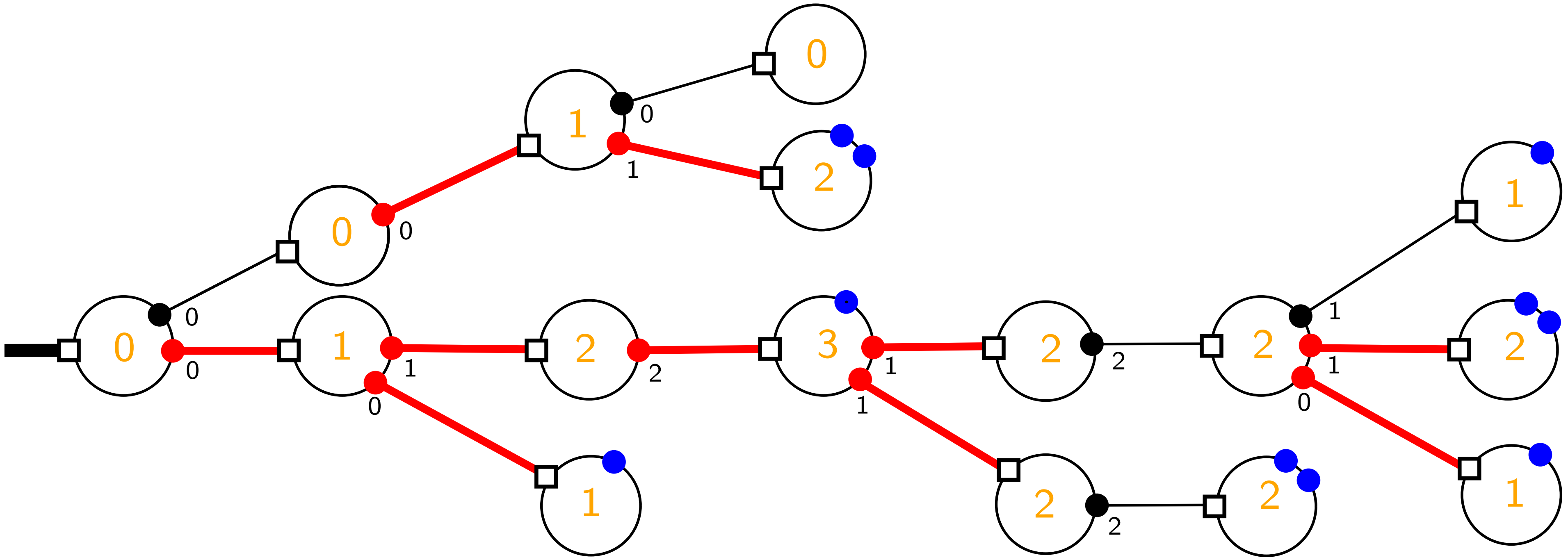
$\bullet$, $\bullet$ are all matched.

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

Observe:

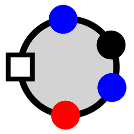
slightly stronger condition than
just asking non negative excess on vertices

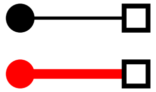
$$\boxed{\text{excess} = \#\{\bullet\} - \#\{\bullet\}}$$



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

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

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



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

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

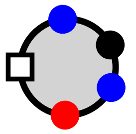
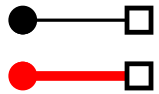
$Q(v, w, u) = \sum_{s\in\mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$ the vertex type gf, where q_s are weights

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** $\}$, $\mathcal{Q} = \Big\{$  $, \ldots \Big\}$  $\bullet$, ${\color{red}\bullet}$ 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

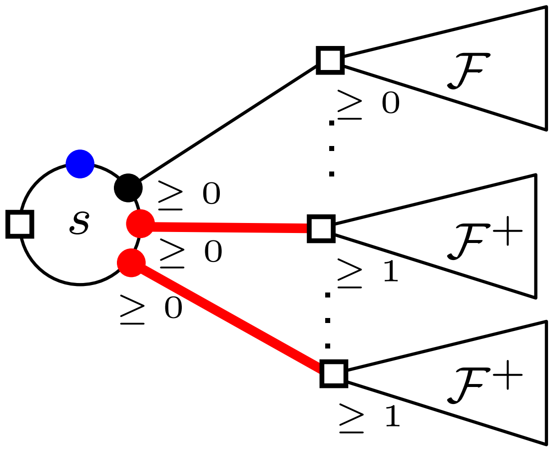
and $F(u) \equiv F(t, u) = \sum_{\tau \in \mathcal{F}} q_\tau t^{|\tau|} u^{\mathrm{excess}(\tau)}$, where $q_\tau = \prod_{s \in \tau} q_s$

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

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

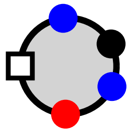
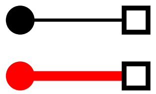
Indeed the equation

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

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

$\bullet$, ${\color{red}\bullet}$ 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 $q_\tau = \prod_{s \in \tau} q_s$

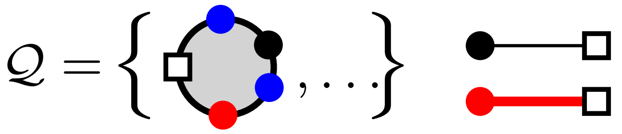
Proposition. The gf $F(u)$ of non negative $\mathcal{Q}$-trees satisfies a catalytic equation of order one:

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

$\Rightarrow$ 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 companion $\mathcal{Q}$-trees

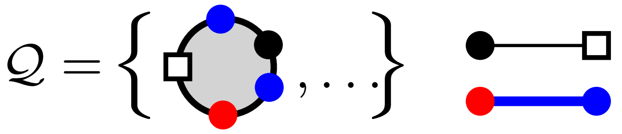
non-negative $\mathcal{Q}$-tree = necklace tree s.t.
- necklaces are in $\mathcal{Q}$
- the excess at each pearl is non negative.



•, • are all matched.

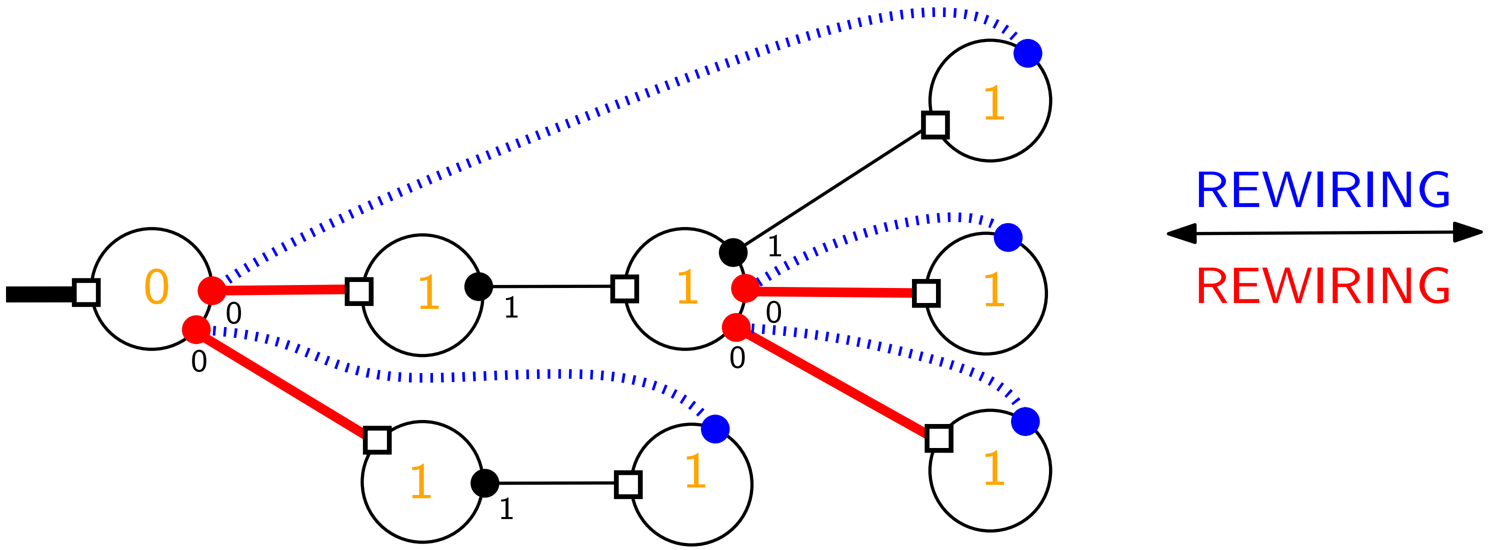
$\#\{\bullet\} \geq \#\{\bullet\}$ in planted subtrees

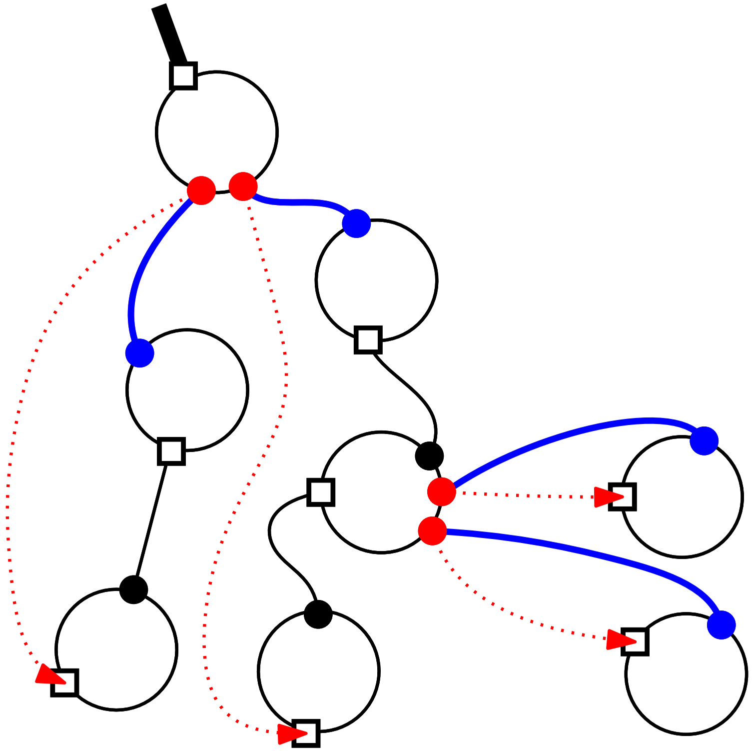
companion $\mathcal{Q}$- tree = necklace tree s.t.
- necklaces are in $\mathcal{Q}$



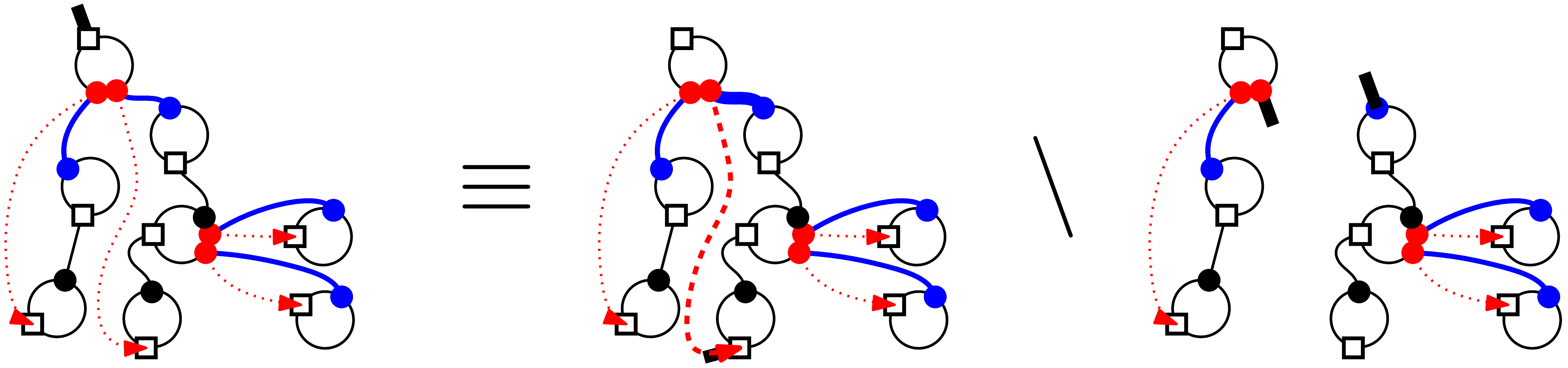
•, •, • are all matched.

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



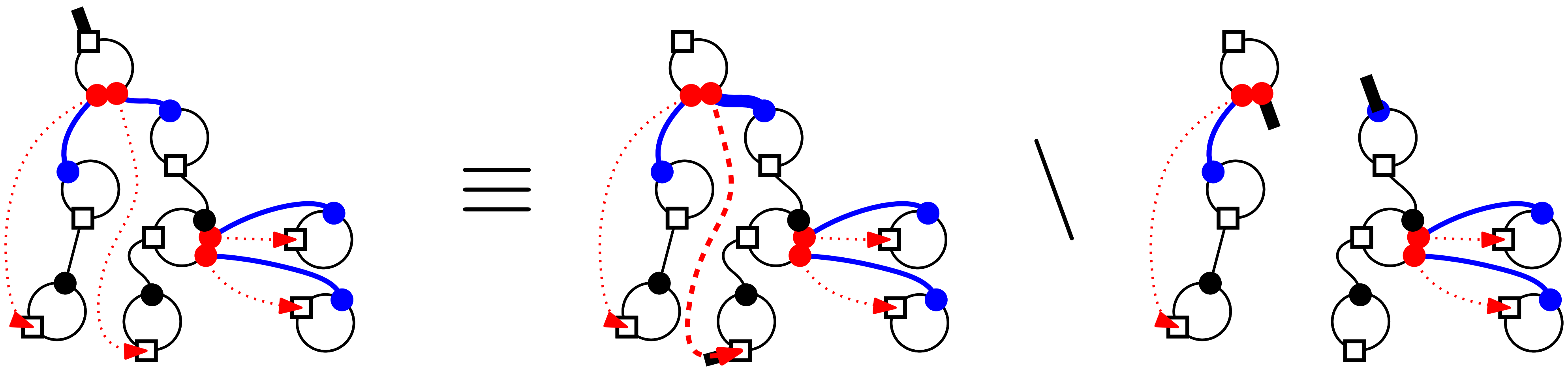


Balanced companion $\mathcal{Q}$-trees VS rooted companion $\mathcal{Q}$-trees

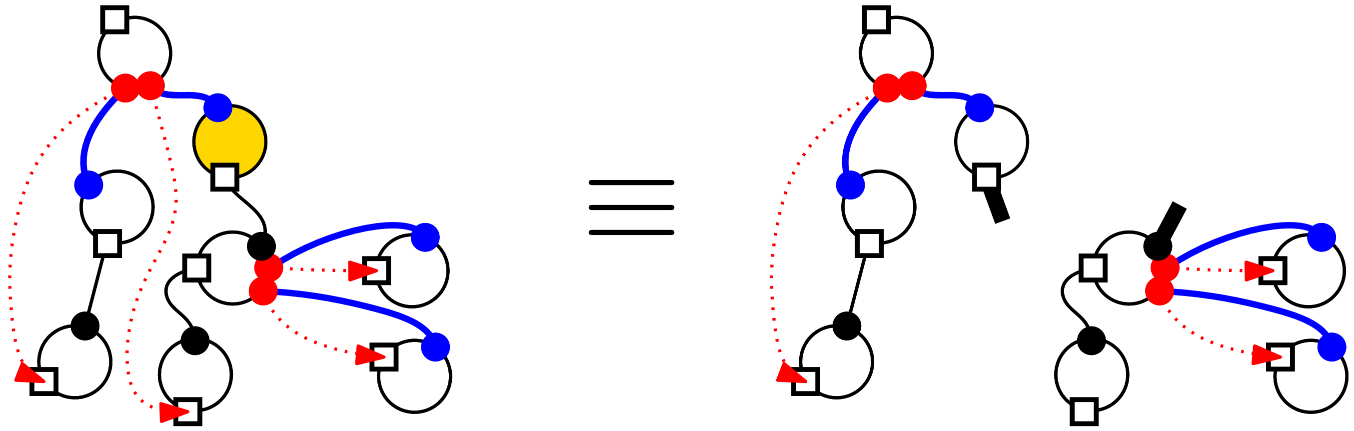


$$\boxed{\mathcal{BC}_{\square} \equiv \mathcal{C}_{\square} - \mathcal{C}_{\bullet} \times \mathcal{C}_{\bullet}}$$

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



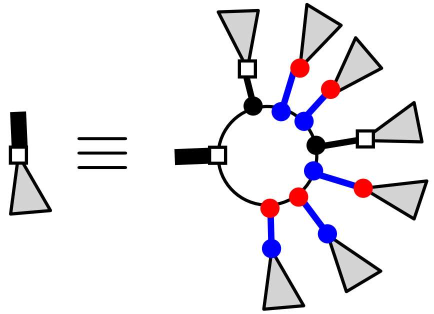
$$\mathcal{BC}_{\square} \equiv \mathcal{C}_{\square} - \mathcal{C}_{\color{red}\bullet} \times \mathcal{C}_{\color{blue}\bullet}$$

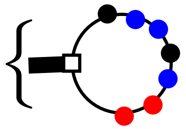


$$\mathcal{C}^{\color{yellow}\circ} \equiv \mathcal{C}_{\square} \times (\varepsilon + \mathcal{C}_{\bullet})$$

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

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

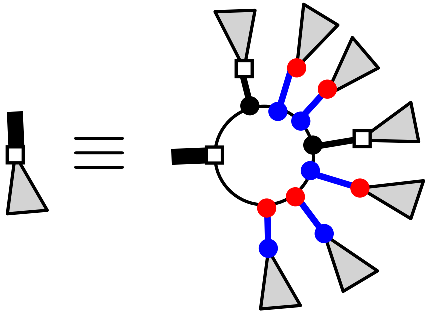


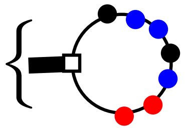
$\mathcal{Q} = \{$  $, \ldots \}$

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

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

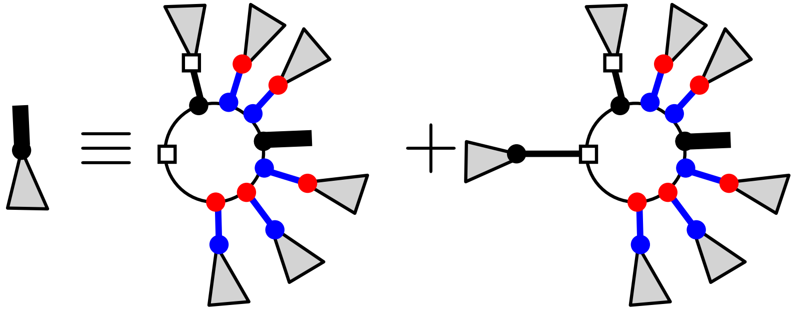
$$C_{\square} = \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

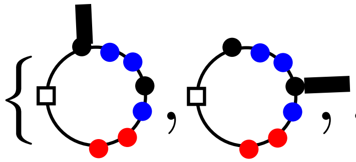


$\mathcal{Q} = \{$  $, \dots \}$

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

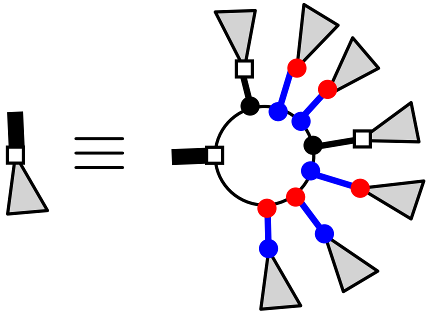
$$C_{\bullet} = \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

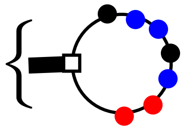


$Q'_{\bullet} = \{$  $, \dots \}$

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

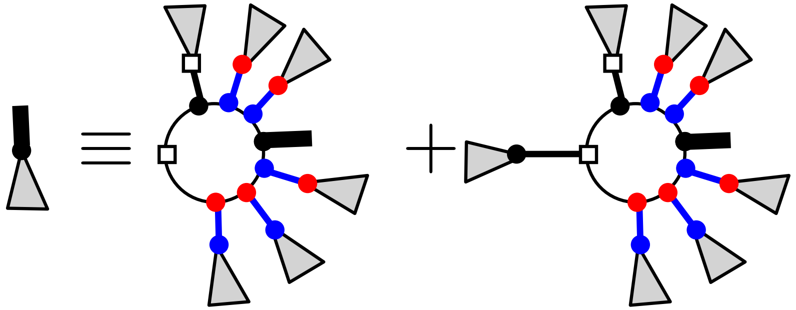
$$C_\square = \mathcal{Z} \times Q(C_\square, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

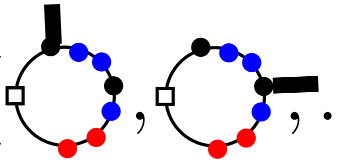


$\mathcal{Q} = \{$  $, \ldots \}$

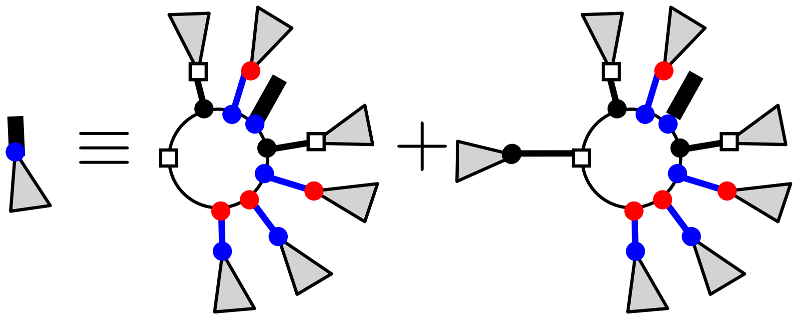
$$Q(v, w, u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$$

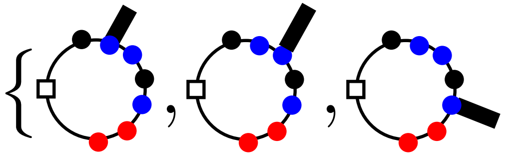
$$C_\bullet = \mathcal{Z} \times (1 + C_\bullet) \times Q'_\bullet(C_\square, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



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

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

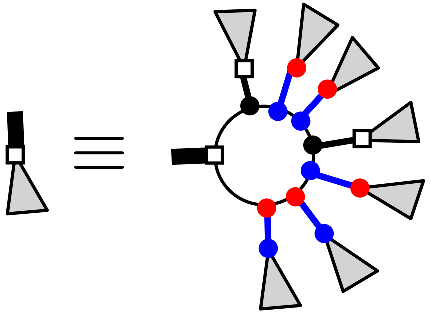


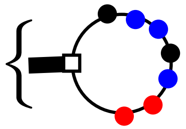
$Q'_{\color{blue}\bullet} = \{$ 

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

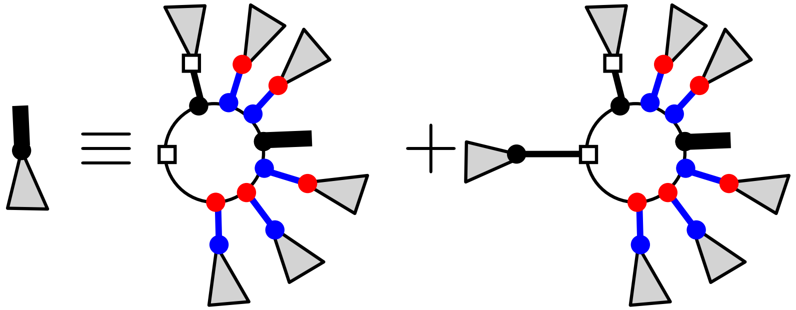
$$C_{\square} \quad = \quad \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$

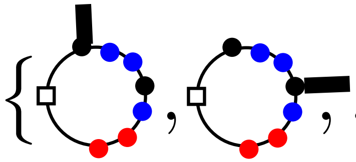
$$Q(v, w, u) = \sum_{s \in \mathcal{Q}} q_s v^{\bullet(s)} w^{{\color{red}\bullet}(s)} u^{{\color{blue}\bullet}(s)}$$



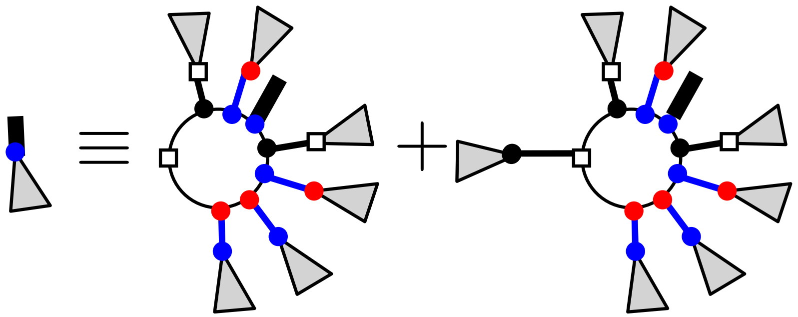
$\mathcal{Q} = \{$  $, \ldots \}$

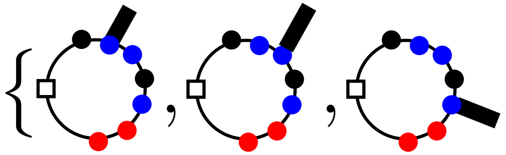
$$C_{\bullet} \quad = \quad \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



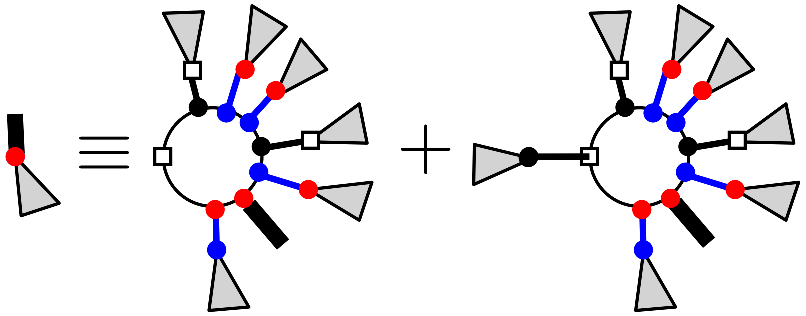
$Q'_{\bullet} = \{$  $, \ldots \}$

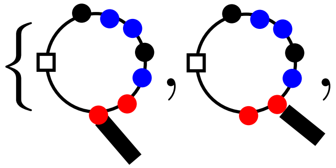
$$C_{\color{blue}\bullet} \quad = \quad \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\color{blue}\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet})$$



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

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



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

The combinatorial lifting of BMJ theorem

THEOREM (Duchi-S. 23)

Let $\mathcal{F} \equiv \mathcal{Z} \times Q\left(\mathcal{F}, \frac{1}{u}(\mathcal{F} \setminus f), u\right)$ 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

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

where the companion trees satisfy:

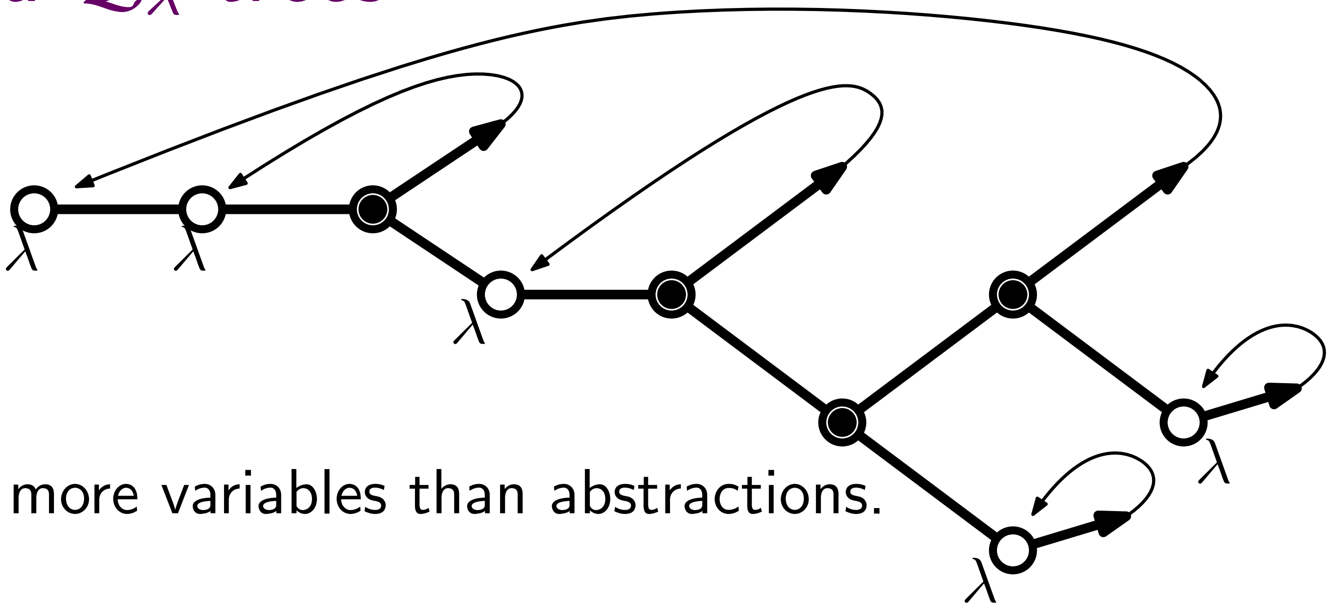
$$\left\{ \begin{array}{lcl} C_{\square} &=& \mathcal{Z} \times Q(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet}) \\ C_{\bullet} &=& \mathcal{Z} \times (1 + C_{\bullet}) \times Q'_{\bullet}(C_{\square}, C_{\color{blue}\bullet}, C_{\color{red}\bullet}) \\ C_{\color{red}\bullet} &=& \mathcal{Z}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

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

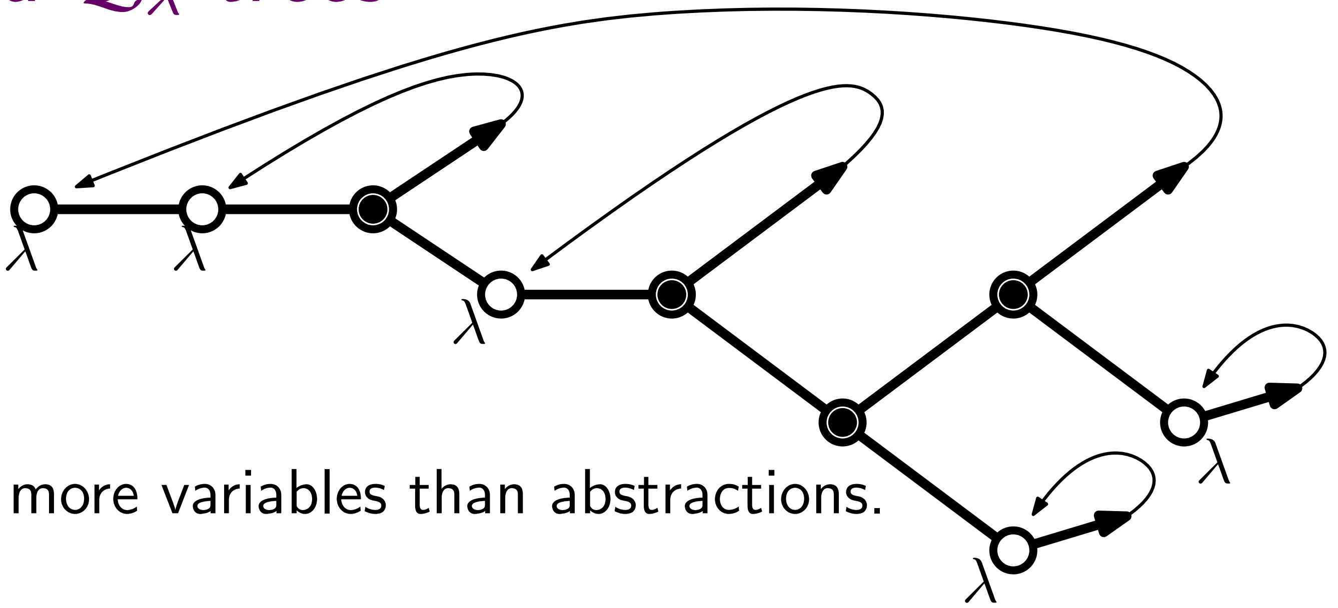
Open planar λ-term are to plane trees with

- 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

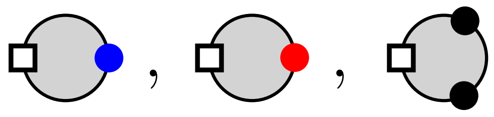


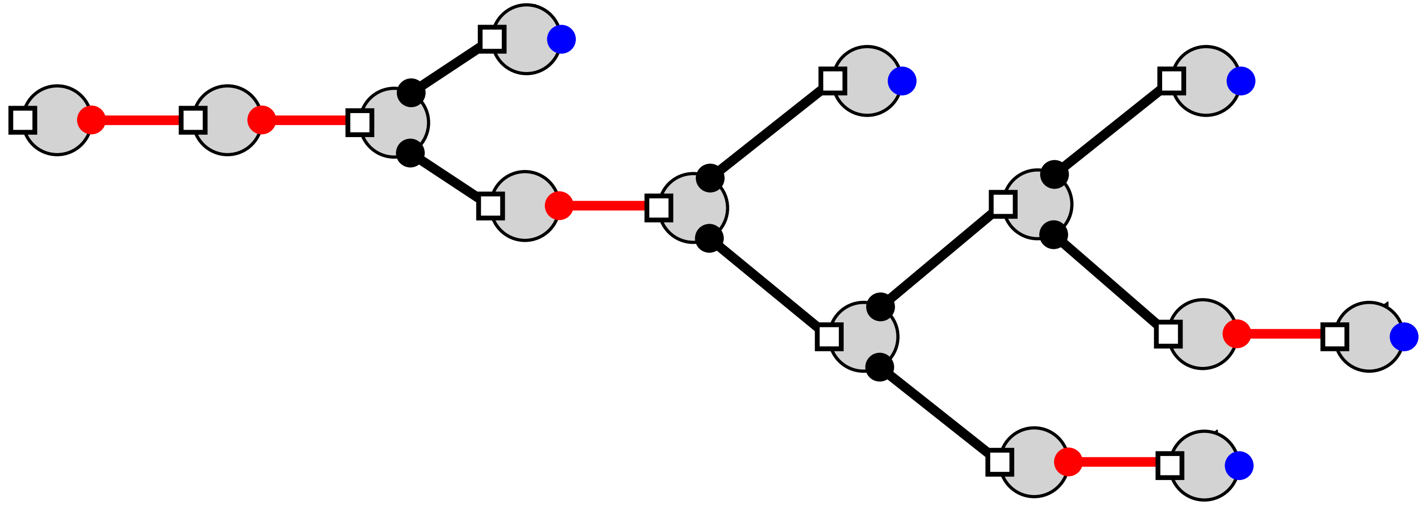
Open planar λ-term are to plane trees with

- 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

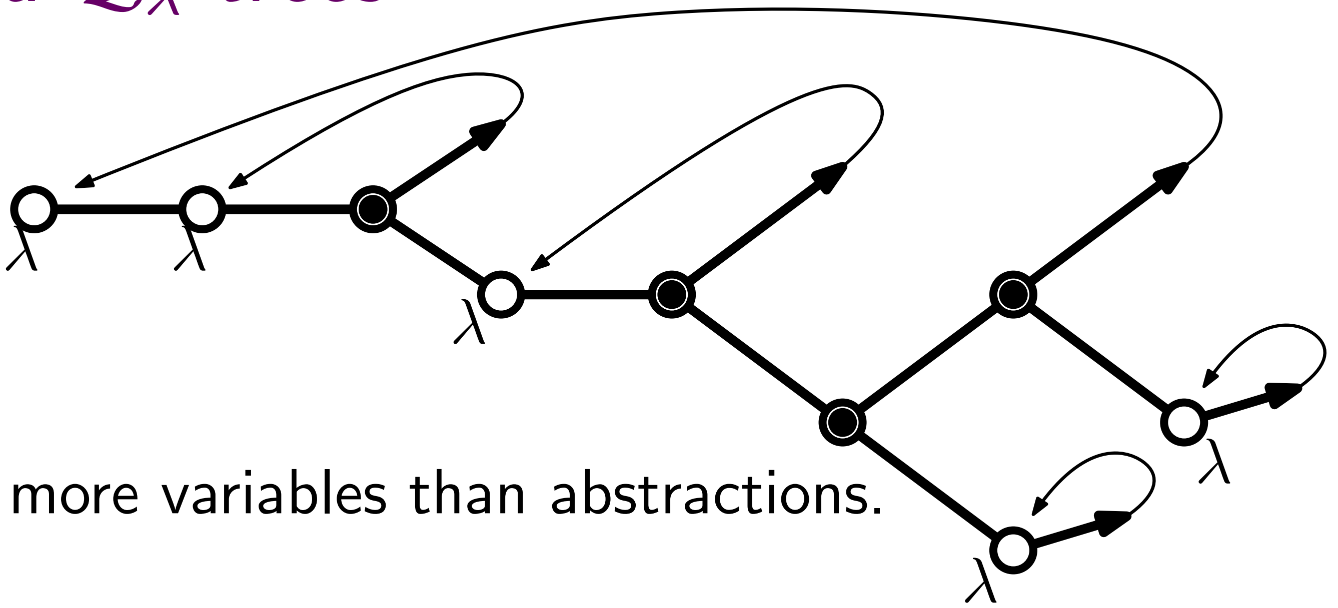
non negative $\mathcal{Q}_\lambda$-trees with excess 0 =
closed planar λ-terms

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

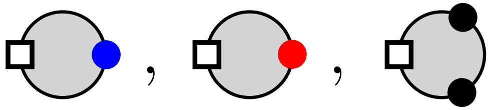
Open planar λ-term are to plane trees with

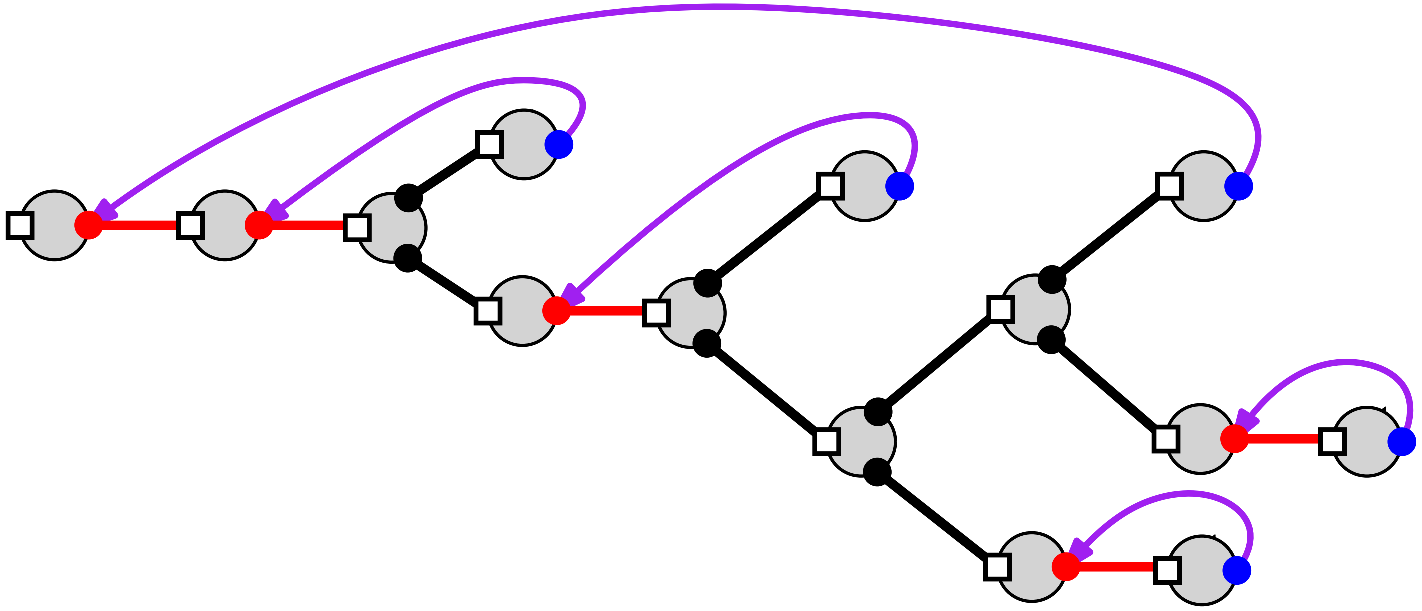
- applications: binary nodes
- abstractions: unary nodes
- variables: leaves, represented as arrow.

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



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

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

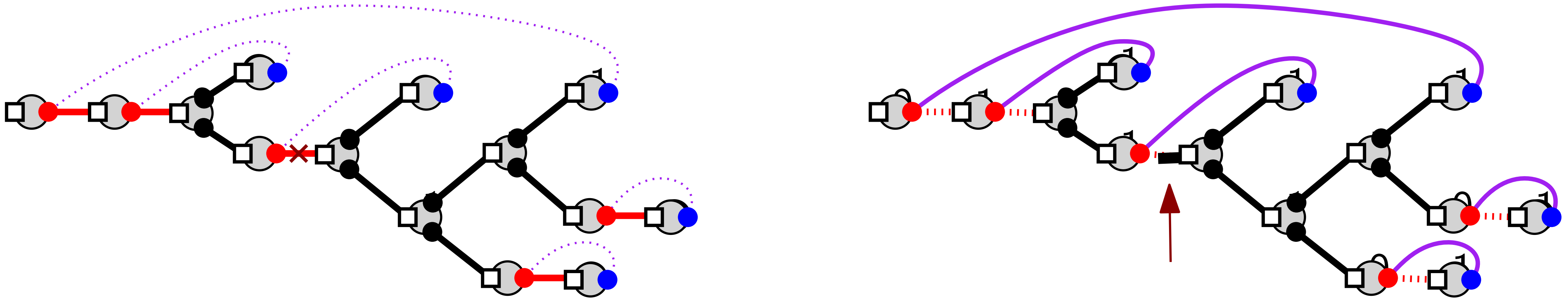


Then **non negative $\mathcal{Q}_\lambda$-trees =**
open planar λ-terms

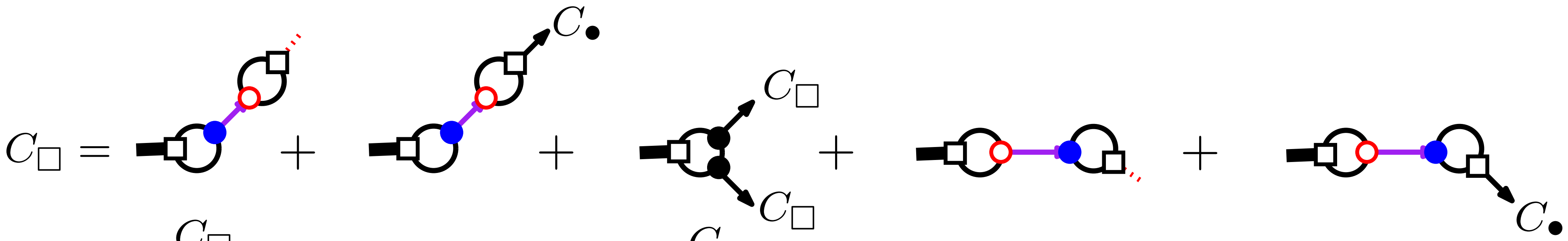
non negative $\mathcal{Q}_\lambda$-trees with excess 0 =
closed planar λ-terms

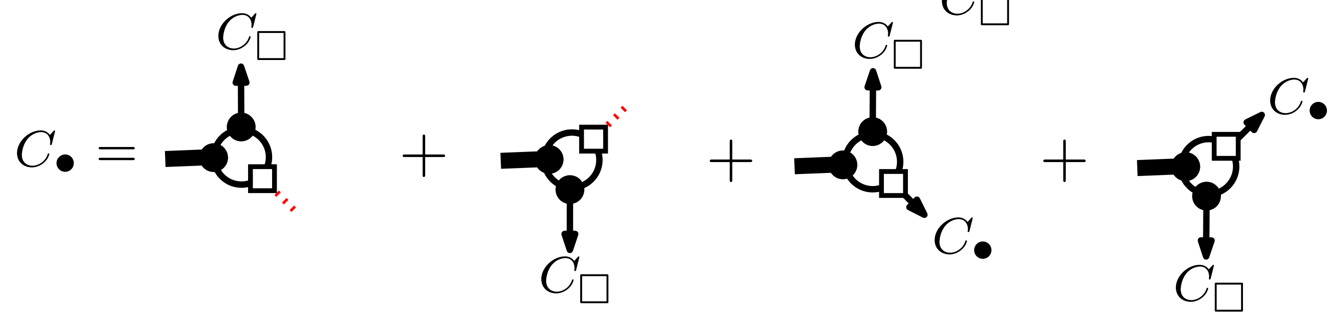
The closure corresponds to the planar abstraction-variable binding.

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)
- ...

Rewiring gives bijections with trees for these models...

$\Rightarrow$ but what are the *pizza slices* for these structures ?

Bijections allow to tackle new parameters...

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

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: I think Emmanuel will indeed agree that

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

$P\,(F(u), f_1, f_2, \ldots, f_k, u, t) = 0$ 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)

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

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

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 !

Thank you,

happy anniversary,

and long life to combinatorial physics !