Symbolic dynamics as a categorical notion

E. Jeandel

Université de Lorraine, France

Plan

Situation

Main open problem of symbolic dynamics:

Decide if two subshifts of finite type are conjugate.

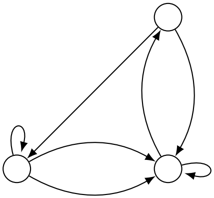
Subshifts of finite type (SFT) can be defined in various ways. Here we focus on the graph approach.

SFTs

Given a finite graph G, the subshift of finite type X_G associated to G is the set of all biinfinite paths on G.

We may think either of G as a graph, or equivalently as a matrix with nonnegative coefficients.

$$\begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$



Conjugacy

- We say that two SFTs are conjugate if the dynamical systems they represent are conjugate.
- If we write the biinfinite paths as words over some infinite alphabet, then the conjugacy is a cellular automaton.

Main problem of symbolic dynamics: decide conjugacy.

Conjugacy

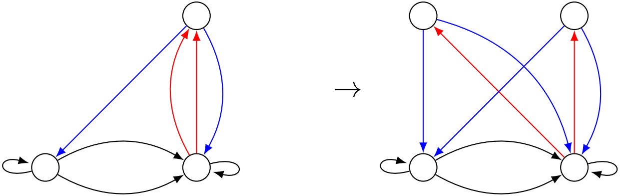
In terms of matrices:

M is *Strong Shift Equivalent* to *N*, if $M \sim N$ where $\sim$ is the smallest equivalence relation s.t. $RS \sim SR$ for all nonsquare integral nonnegative matrices R, S

In terms of graph:

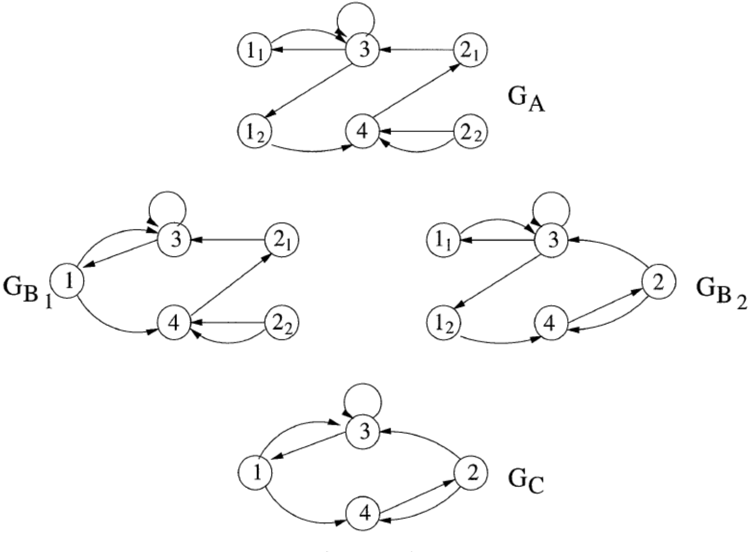
G is conjugate to *G'* if *G* can be obtained from *G'* by a series of incoming/outgoing splits and amalgamations.

Incoming split: transform one vertex u into two vertices u_1, u_2, split the inputs and share the outputs.



Examples

All pictures from Kitchen's book (Symbolic Dynamics):



Examples

All pictures from Kitchen's book (Symbolic Dynamics):

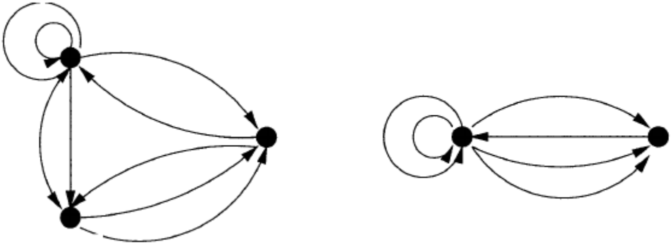


Figure 2.1.5

Examples

All pictures from Kitchen's book (Symbolic Dynamics):

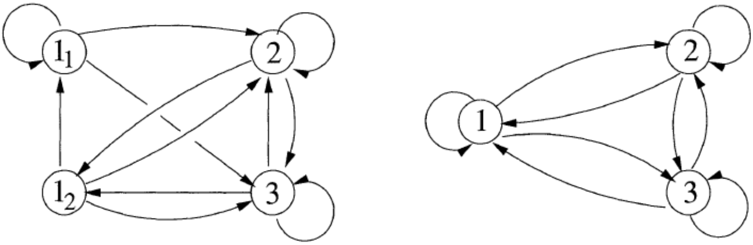


Figure 2.1.6

History

I will use “Strong Shift equivalence” (SSE) instead of conjugacy

- Williams 1973: SSE is introduced
- Williams 1973: SSE is decidable for one-sided SFTs (only incoming splits/amalgamations)
- Franks 1984: Flow equivalence (a variant of SSE) is decidable
- Kim-Roush 1988: Shift equivalence (a variant of SSE) is decidable
- Kim-Roush 1992: Shift equivalence is not the same as SSE
- Folklore: SSE is decidable for matrices in $\mathbb{Z}$ rather than in $\mathbb{Z}_+$ (graphs with negative edges)

Conclusion: while SSE is not known to be decidable, there are a lot of variants that are.

This talk

- SSE is complicated because the split/amalgamation stuff is complicated
- We will introduce a simplified version of the split/amalgamation
- The equations we obtain will remind us of category theory, and we will use category theory to obtain some results

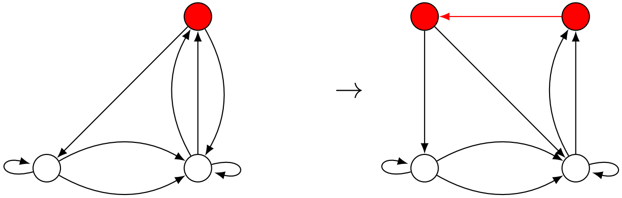
Plan

Flow equivalence (Parry-Sullivan 1975)

We will first focus on *flow equivalence*, a variant of SSE.

Flow equivalence is just SSE with a looser notion of time

i.e. we can now stretch a vertex:



Plan

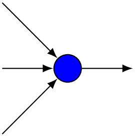
- We will reformulate flow equivalence with simpler equations
- Then we will go back to the original problem

Goal: get rid of the split/amalgamations equations.

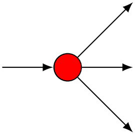
Idea

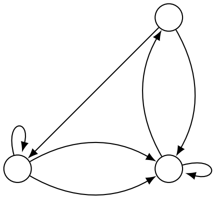
Represent the graph in a new formalism with two kinds of vertices:

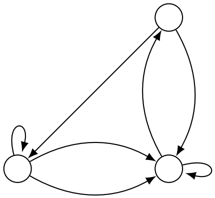
- Vertices that collect incoming edges

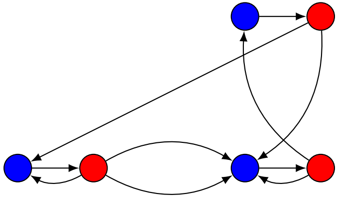


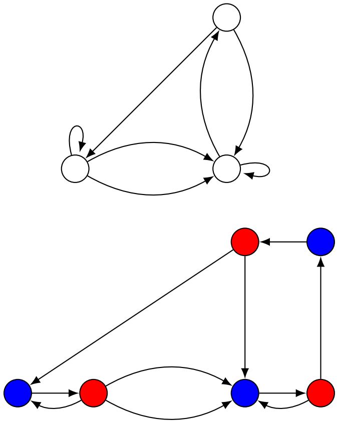
- Vertices that distribute outgoing edges:











Rules

How does flow equivalence translate into rules for red-blue graphs ?

Rules

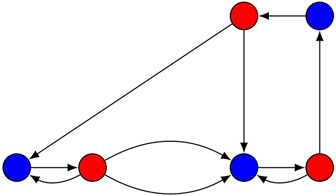
We want to think of the blue vertex as gathering incoming edges:

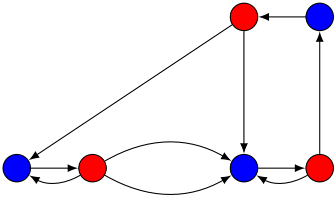
- Gathering one incoming edge is the same as doing nothing
- Gathering three incoming edges is the same as gathering the first two, then gathering the result with the third

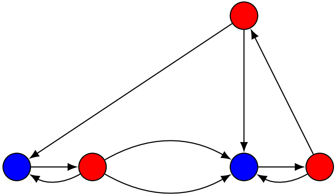
We only need blue vertices of incoming degree 2

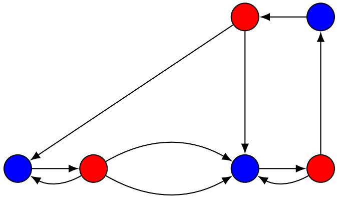
(Technically we also need vertices of incoming degree 0)

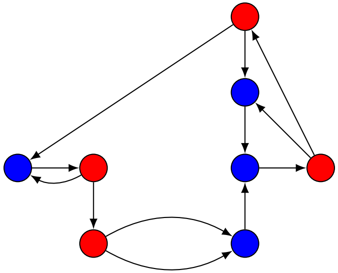
The same is true for red vertices

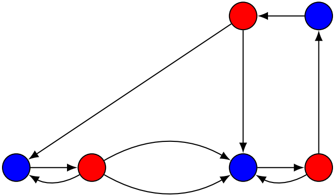


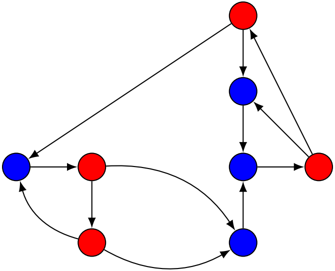


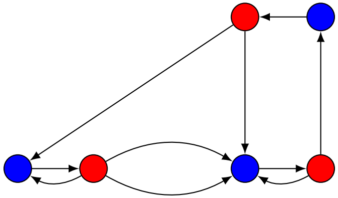


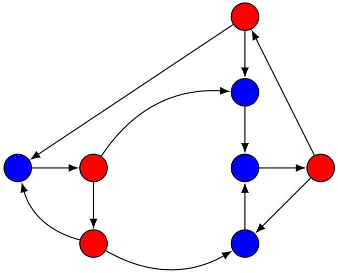




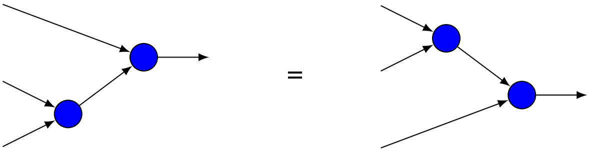




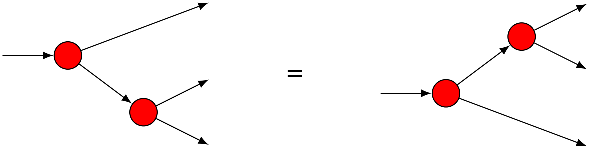




Two rules



Two rules

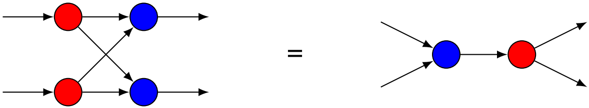


Axioms

What axioms do we need to take into account amalgamations/split ?

Axioms

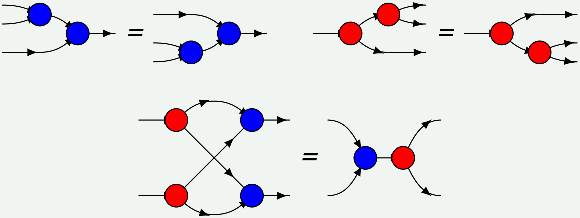
We only need ONE additional axiom:



Theorem 1

Theorem

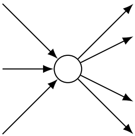
Flow equivalence, when expressed on bicolored graphs is entirely given by the following equations;



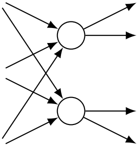
(plus other axioms for degenerate graphs, i.e. graphs with sources and sinks)

Idea of the proof

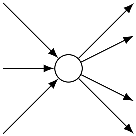
As an example, how to do the following split?



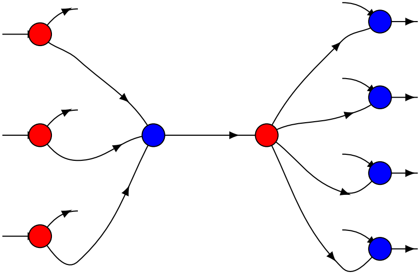
=



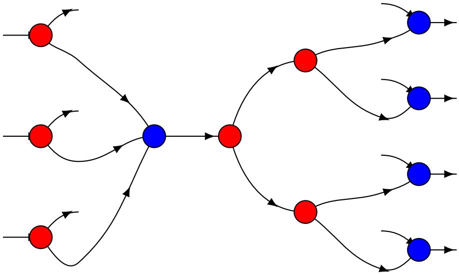
Proof



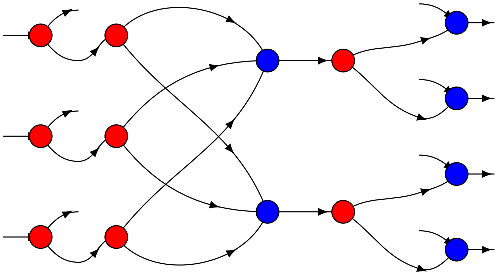
Proof



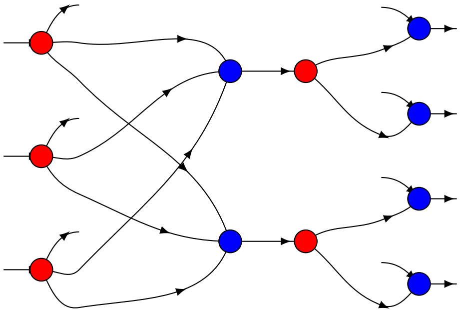
Proof



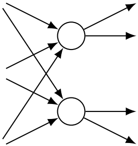
Proof



Proof



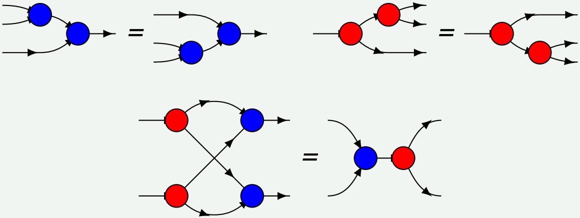
Proof



Theorem 1

Theorem

Flow equivalence, when expressed on bicolored graphs is entirely given by the following equations;



(plus other axioms for degenerate graphs, i.e. graphs with sources and sinks)

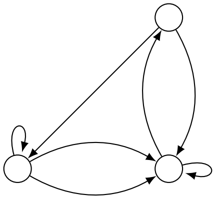
Strong Shift Equivalence

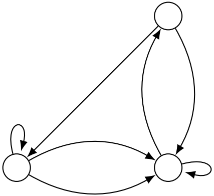
How to go back to strong shift equivalence (conjugacy) ?

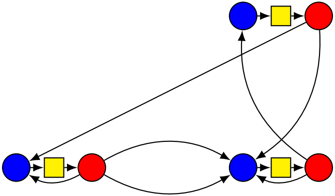
Flow equivalence is just SSE with a looser notion of time

SSE is just flow equivalence with a stronger notion of time.

(formal statement uses results from Boyle and Wagoner)
We will add a new vertex that represents one unit of time



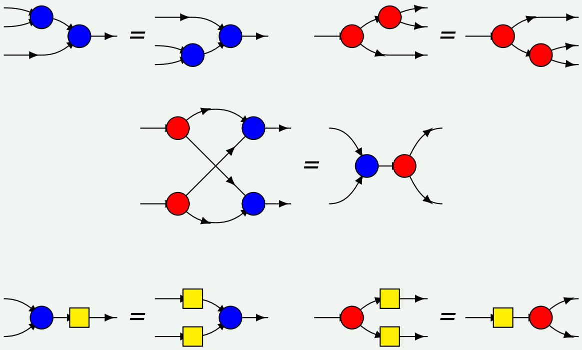




Theorem 2

Theorem

SSE, when expressed on bicolored graphs is entirely given by the following equations;

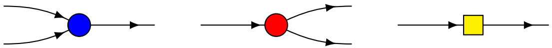


(plus other axioms for degenerate graphs)

Plan

Categories

Idea: Do not see these boxes as nodes in a graph, but as operators :



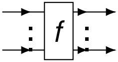
Typically, the blue node takes two inputs, and converts them to one output, similarly for the others.
What do we need to represent graphs ?

- A way to compose these operators sequentially
- A way to compose these operators in parallel

What we need is a symmetric monoidal category.

Categories

A *prop* is the data, for each pair (n, m) of a set $P[m, n]$.
Think of elements of $P[m, n]$ as boxes with m inputs and n outputs. We write $f : m \to n$.

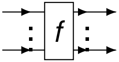


We also need :

- A composition $P[n, p] \times P[m, n] \to P[m, p]$ satisfying the obvious axioms.
- An identity element:
- A tensor product : $P[m_1, n_1] \times P[m_2, n_2] \to P[m_1 + m_2, n_1 + n_2]$ satisfying the obvious axioms
- A swap element:

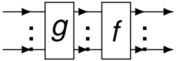
Categories

A *prop* is the data, for each pair (n, m) of a set $P[m, n]$.
Think of elements of $P[m, n]$ as boxes with m inputs and n outputs. We write $f : m \to n$.



We also need :

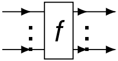
- A composition $P[n, p] \times P[m, n] \to P[m, p]$ satisfying the obvious axioms.

  $f \circ g$: 

- An identity element:
- A tensor product : $P[m_1, n_1] \times P[m_2, n_2] \to P[m_1 + m_2, n_1 + n_2]$ satisfying the obvious axioms
- A swap element:

Categories

A *prop* is the data, for each pair (n, m) of a set $P[m, n]$.
Think of elements of $P[m, n]$ as boxes with m inputs and n outputs. We write $f : m \to n$.



We also need :

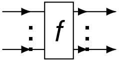
- A composition $P[n, p] \times P[m, n] \to P[m, p]$ satisfying the obvious axioms.
- An identity element:

  id: 

- A tensor product : $P[m_1, n_1] \times P[m_2, n_2] \to P[m_1 + m_2, n_1 + n_2]$ satisfying the obvious axioms
- A swap element:

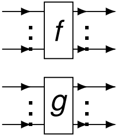
Categories

A *prop* is the data, for each pair (n, m) of a set $P[m, n]$.
Think of elements of $P[m, n]$ as boxes with m inputs and n outputs. We write $f : m \to n$.



We also need :

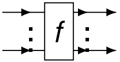
- A composition $P[n, p] \times P[m, n] \to P[m, p]$ satisfying the obvious axioms.
- An identity element:
- A tensor product : $P[m_1, n_1] \times P[m_2, n_2] \to P[m_1 + m_2, n_1 + n_2]$ satisfying the obvious axioms

$f \otimes g$ 

- A swap element:

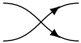
Categories

A *prop* is the data, for each pair (n, m) of a set $P[m, n]$.
Think of elements of $P[m, n]$ as boxes with m inputs and n outputs. We write $f : m \to n$.



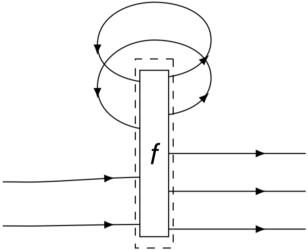
We also need :

- A composition $P[n, p] \times P[m, n] \to P[m, p]$ satisfying the obvious axioms.
- An identity element:
- A tensor product : $P[m_1, n_1] \times P[m_2, n_2] \to P[m_1 + m_2, n_1 + n_2]$ satisfying the obvious axioms
- A swap element:

σ: 

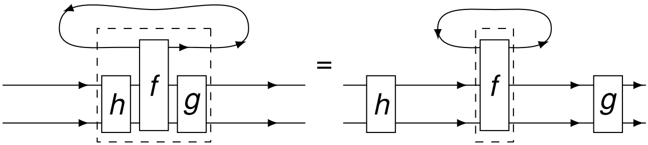
Categories

A *traced prop* is a prop that contains an operator:
$[n+1, m+1] \to [n, m]$, called the trace satisfying obvious axioms



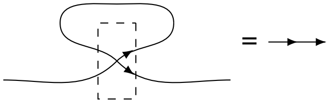
Categories

A *traced prop* is a prop that contains an operator:
$[n+1, m+1] \to [n, m]$, called the trace satisfying obvious axioms



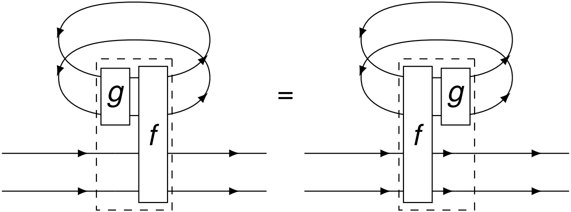
Categories

A *traced prop* is a prop that contains an operator:
$[n + 1, m + 1] \to [n, m]$, called the trace satisfying obvious axioms



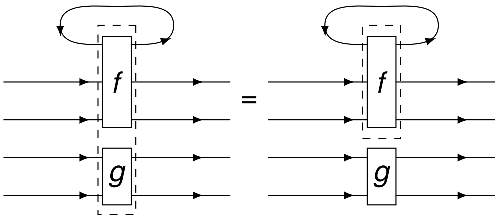
Categories

A *traced prop* is a prop that contains an operator: $[n+1, m+1] \to [n, m]$, called the trace satisfying obvious axioms



Categories

A *traced prop* is a prop that contains an operator:
$[n+1, m+1] \to [n, m]$, called the trace satisfying obvious axioms



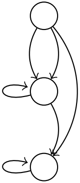
Main idea

- Find a traced prop which contains a *bigebra*, that is
  - An element $2 \to 1$ to represent the blue node
  - An element $1 \to 2$ to represent the red node
  - An arrow $1 \to 1$ to represent the square
- Suppose these three things satisfy the axioms we gave previously
- Then one can “interpret” graphs/matrice/SFTs in this category in such a way that SFTs that are conjugate corresponds to the same element of the prop.
- This gives a way to obtain *invariants*

Main idea

- Start from a graph/matrix

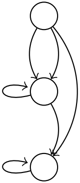
$$M = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$



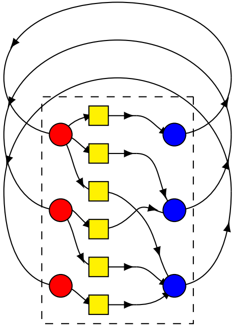
Main idea

- Start from a graph/matrix

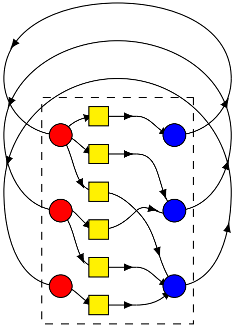
$$M = \begin{pmatrix} 0 & 2 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$



- Convert it into a red/blue graph:



Main idea

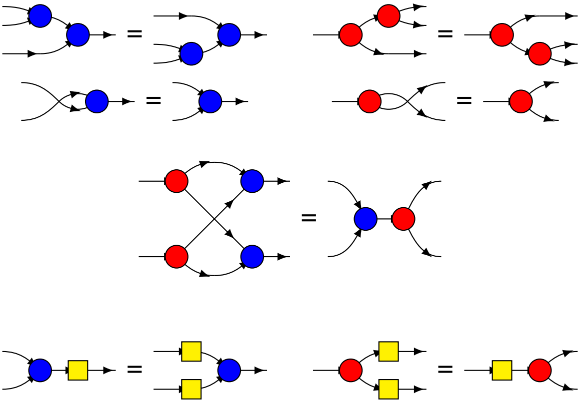
- Convert it into a red/blue graph: 

- Interpret the nodes as operators in some category:

54

The equations

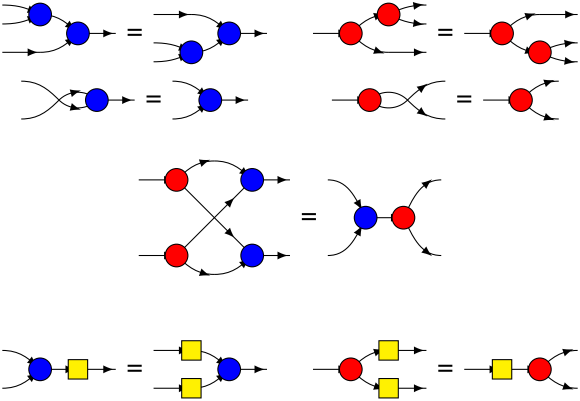
These equations are incredibly common, and appear in many parts of math:



Plan

The equations

These equations are incredibly common, and appear in many parts of math:



Plan

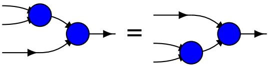
1. Introduction
2. The simplification
3. Categories
4. Examples
   - Monoids
   - Bialgebras and Hopf Algebras
   - Groups
5. Conclusion

Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:

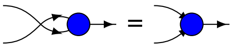
Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:



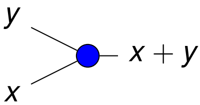
Monoids

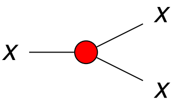
Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:

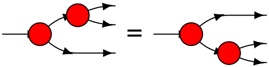


Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:

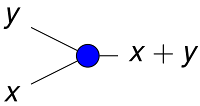


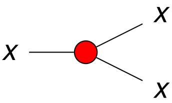


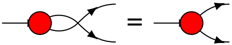


Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:

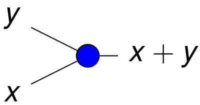


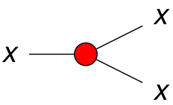


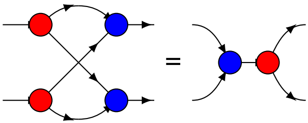


Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:

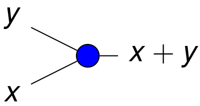


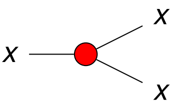




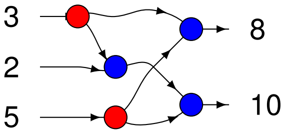
Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:



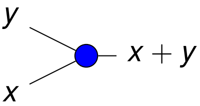


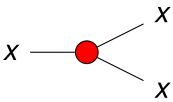
$\mathcal{M} = (\mathbb{R}, +)$:



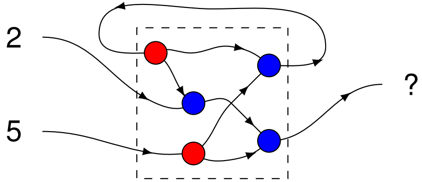
Monoids

Let $\mathcal{M}$ be a commutative monoid. Inputs and outputs are elements of $\mathcal{M}$:





$\mathcal{M} = (\mathbb{R}, +)$:



Monoids

Solution: monoids with multiplicities:

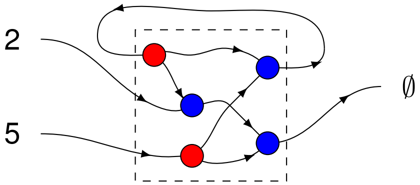
- Input of size *n*: an element of $\mathcal{M}^n \to \mathbb{N}_\infty$.
- The trace counts for how many elements of $\mathcal{M}$ the diagram makes sense.

Monoids

Solution: monoids with multiplicities:

- Input of size n: an element of $\mathcal{M}^n \to \mathbb{N}_\infty$.
- The trace counts for how many elements of $\mathcal{M}$ the diagram makes sense.

$\mathcal{M} = (\mathbb{R}, +)$:

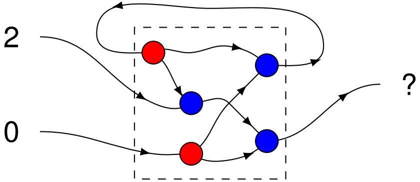


Monoids

Solution: monoids with multiplicities:

- Input of size *n*: an element of $\mathcal{M}^n \to \mathbb{N}_\infty$.
- The trace counts for how many elements of $\mathcal{M}$ the diagram makes sense.

$\mathcal{M} = (\mathbb{R}, +)$:

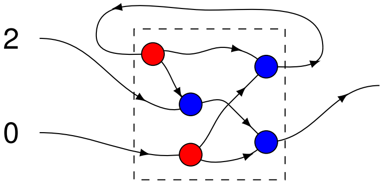


Monoids

Solution: monoids with multiplicities:

- Input of size n: an element of $\mathcal{M}^n \to \mathbb{N}_\infty$.
- The trace counts for how many elements of $\mathcal{M}$ the diagram makes sense.

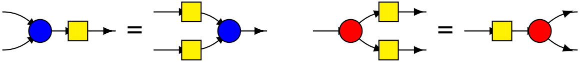
$\mathcal{M} = (\mathbb{R}, +)$:



all elements of $\mathbb{R}$

Monoids

What about the square ?



It's just a morphism for the monoid (which will automatically work with the copy)

Theorem

Theorem

Let R be a matrix and $\mathcal{M}$ be a monoid, and h an homomorphism. When interpreting the diagram in the previous category, R represents the number of solutions of the equation $x = h(Rx)$ in the monoid $\mathcal{M}$.

Theorem

For all commutative monoids $\mathcal{M}$ and all homomorphisms h of $\mathcal{M}$, the number of solutions of the equation $x = h(Rx)$ in $\mathcal{M}$ is an invariant of conjugacy.

Plan

Bialgebras

Bialgebras and Hopf Algebras are well studied in representation theory and combinatorics.

- Input of size n: an element of $V^{\otimes n}$ where V is a vector space over some field $\mathbb{K}$
- If V is a vector space with basis e_i, $V \otimes V$ is a vector space with basis $e_i \otimes e_j$
- Boxes are linear maps

Monoid ring

Monoid ring : $\mathbb{K}[\mathcal{M}]$, vector space with basis e_x, $x \in \mathcal{M}$

- Multiplication: $e_x \otimes e_y \to e_{x+y}$
- By the multiplication:
  $3(e_2 \otimes e_3) - 2(e_1 \otimes e_4) + 3(e_1 \otimes e_5) \to e_5 + 3e_6$
- Comultiplication $e_x \to e_x \otimes e_x$
- By the comultiplication: $e_5 + 3e_6 \to e_5 \otimes e_5 + 3e_6 \otimes e_6$

Exactly the same example as before, presented differently.

The binomial bialgebra

The binomial bialgebra: $V = \mathbb{K}[X]$, basis $(X^n)_{n \geq 0}$

- Multiplication: $X^n \otimes X^m \to X^{n+m}$
- By the multiplication:
  $3(X^2 \otimes X^3) - 2(X^1 \otimes X^4) + 3(X^1 \otimes X^5) \to X^5 + 3X^6$
- Comultiplication $X^n \to \sum_k \binom{n}{k} X^k \otimes X^{n-k}$
- By the comultiplication: $X^2 \to 1 \otimes X^2 + 2X \otimes X + X^2 \otimes 1$
- Homomorphism: $X^n \to (\lambda X)^n$ for some $\lambda \in \mathbb{K}$

The binomial bialgebra

The canonical example $V = \mathbb{K}[X]$ does not have a trace, we need to tweak it:

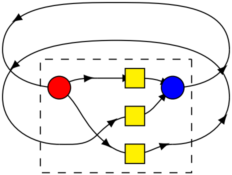
- Coefficients in the complete semiring $\mathbb{R}_\infty$ rather than in $\mathbb{R}$
- We allow infinite sums: $V = \mathbb{R}_\infty[[X]]$

Trace: sum over all n of the coefficient of X^n of the output if the input is X^n

Example

The golden shift:

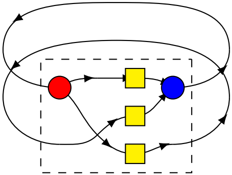
$$M = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$



Example

The golden shift:

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$



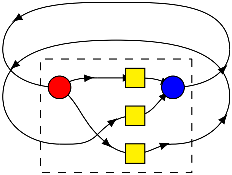
We look without the traces.
If we start from $X^n \otimes X^m$, the output is

$$\lambda^{n+m} \sum_k \binom{n}{k} X^{m+k} \otimes X^{n-k}$$

Example

The golden shift:

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$



We look without the traces.
If we start from $X^n \otimes X^m$, the output is

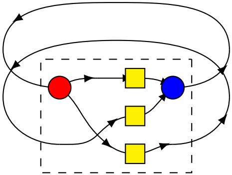
$$\lambda^{n+m} \sum_k \binom{n}{k} X^{m+k} \otimes X^{n-k}$$

The coefficient of $X^n \otimes X^m$ in this sum is $\binom{n}{n-m} \lambda^{n+m}$

Example

The golden shift:

$$M = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$



The coefficient of $X^n \otimes X^m$ in this sum is $\binom{n}{n-m}\lambda^{n+m}$
The value of the graph is therefore

$$\sum_{n,m} \binom{n}{n-m} \lambda^{n+m} = \frac{1}{1-\lambda^2-\lambda}$$

Zeta function

Theorem

Let M be a nonnegative matrix.
The result of the computation is $\zeta_M(\lambda)$, with $\zeta_M(t) = \frac{1}{\det(I - tM)}$.
Therefore ζ_M is an invariant of conjugacy.

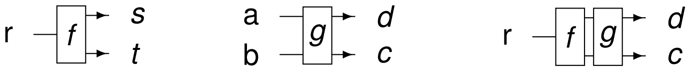
Consequence of McMahon master's theorem.

Plan

Cospans

New, weird category:

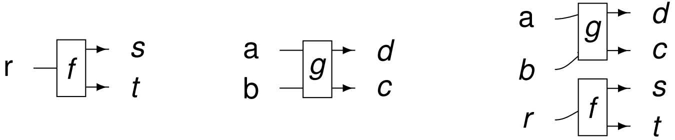
- A box with *n* inputs and *m* outputs is a *commutative group*, with at least $n+m$ generators, and a finite presentation.
- Inputs and outputs are to be understood as generators that can still be plugged in into other generators
- Composition is the new group obtained by identifying input and output generators that are plugged together (pushout)



$$\left\langle \begin{matrix} r,s, \\ t,u \end{matrix} \;\middle|\; u=2t+r \right\rangle \quad \left\langle \begin{matrix} a,b, \\ c,d \end{matrix} \;\middle|\; \begin{matrix} a-b=c+d \\ c-3d=a \end{matrix} \right\rangle \quad \left\langle \begin{matrix} a,b, \\ c,d, \\ r,s, \\ t,u \end{matrix} \;\middle|\; \begin{matrix} a-b=c+d \\ u=2t+r \\ c-3d=a \\ s=a \\ t=b \end{matrix} \right\rangle$$

Groups

- Tensor product is the new group obtained by putting the two groups side by side (sum of the group)



$$\left\langle \begin{matrix} r,s, \\ t,u \end{matrix} \;\middle|\; u=2t+r \right\rangle \quad \left\langle \begin{matrix} a,b, \\ c,d \end{matrix} \;\middle|\; \begin{matrix} a-b=c+d \\ c-3d=a \end{matrix} \right\rangle \quad \left\langle \begin{matrix} a,b, \\ c,d, \\ r,s, \\ t,u \end{matrix} \;\middle|\; \begin{matrix} a-b=c+d \\ u=2t+r \\ c-3d=a \end{matrix} \right\rangle$$

Groups

- Trace consists in equating input and output
- We have to look at groups upto isomorphism of the internal generators.

What is the red and blue node ?

- Red node: group $\mathbb{Z} = \langle x, y, z | x = y = z \rangle$: all generators are equal
- Blue node: group $\mathbb{Z}^2 = \langle x, y, z | x + y = z \rangle$: output generator is equal to the sum of the input generators.

Note: the square is the trivial homomorphism

Theorem

Theorem

Starting from a matrix M (or a graph G), this construction associates to M the abelian group

$$G = \langle x | x = Mx \rangle$$

This is the Bowen-Franks group

Theorem

We can do the same with things other than groups: if we look at $\mathbb{Z}[t]$-modules instead of groups, we can have a nontrivial interpretation of the square, and obtain:

Theorem

Starting from a matrix M (or a graph G), this construction associates to M the $\mathbb{Z}[t]$ module:

$$G = \langle x | x = tMx \rangle$$

This is the dimension group (Krieger).

Plan

Conclusion

A systematic way to obtain invariants for symbolic dynamics by looking at algebraic structures in some categories.
We recover the classical invariants, which proves the method works:

- The Zeta function
- The Bowen-Franks group
- The Dimension group

Now: test other categories, to obtain new invariants!