Gershom Bazerman

Homological Computations for Term Rewriting Systems

Papers We Love, NY
Aug 2017

Homological Computations for Term Rewriting Systems

Philippe Malbos[1] and Samuel Mimram[2]

1 LIX, École Polytechnique
samuel.mimram@lix.polytechnique.fr
2 ICJ, Université de Lyon
malbos@math.univ-lyon1.fr

Abstract

An important problem in universal algebra consists in finding presentations of algebraic theories by generators and relations, which are as small as possible. Exhibiting lower bounds on the number of those generators and relations for a given theory is a difficult task because it a priori requires considering all possible sets of generators for a theory and no general method exists. In this article, we explain how homological computations can provide such lower bounds, in a systematic way, and show how to actually compute those in the case where a presentation of the theory by a convergent rewriting system is known. We also introduce the notion of coherent presentation of a theory in order to consider finer homotopical invariants. In some aspects, this work generalizes, to term rewriting systems, Squier's celebrated homological and homotopical invariants for string rewriting systems.

FSCD 2016

FSCD 2016 ▾ Submission ▾ Programme ▾ Registration Travel & Local Info ▾

COMMITTEES

Organization Committee | Program Committee | Steering Committee

FSCD STEERING COMMITEE:

Thorsten Altenkirch (University Nottingham)

Gilles Dowek (INRIA)

Santiago Escobar (University Politecnica de Valencia)

Maribel Fernandez (King's College London)

Masahito Hasegawa (University Kyoto)

Hugo Herbelin (INRIA)

Nao Hirokawa (JAIST)

Luke Ong (Chair, University Oxford)

Jens Palsberg (UCLA)

Kristoffer Rose (Two Sigma Investments)

Rene Thiemann (University Innsbruck)

Pawel Urzyczyn (University Warsaw)

Femke van Raamsdonk (VU University Amsterdam)

Homological Computations

~~for Term Rewriting Systems~~

Polyhedron		Vertices	Edges	Faces
tetrahedron		4	6	4
cube		8	12	6
octahedron		6	12	8
dodecahedron		20	30	12
icosahedron		12	30	20

$4 - 6 + 4 = 2$

$8 - 12 + 6 = 2$

$6 - 12 + 8 = 2$

$20 - 30 + 12 = 2$

$12 - 30 + 20 = 2$

(a) Homology (theory) is a Functor

Mathematical Object (like a space)

->

Sequence of Mathematical Objects (like groups)

An Aside on Groups

- A set with a single associative operation (•), a zero element (e), and a negation operation such that a • -a = e.
- A generating set with terms as sequences of elements of the set, zero, and their negations under the group laws, and an identification of some terms (e.g. adq=bc).
- A closed collection of permutations of a set (Cayley).
- A one object category with all morphisms invertible
- Closed paths in a space.

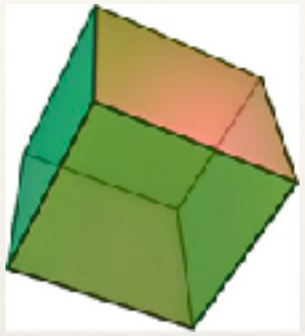
An Aside on Groups

- **A one object category with all morphisms invertible**

  Since categories are considered up to isomorphism, this is *the* group. In all other cases there may be multiple descriptions which map, one to one, to one another.

  The *rank* of a group is the size of the *smallest* generating set of the group.

(a) Homology (theory) is a Functor

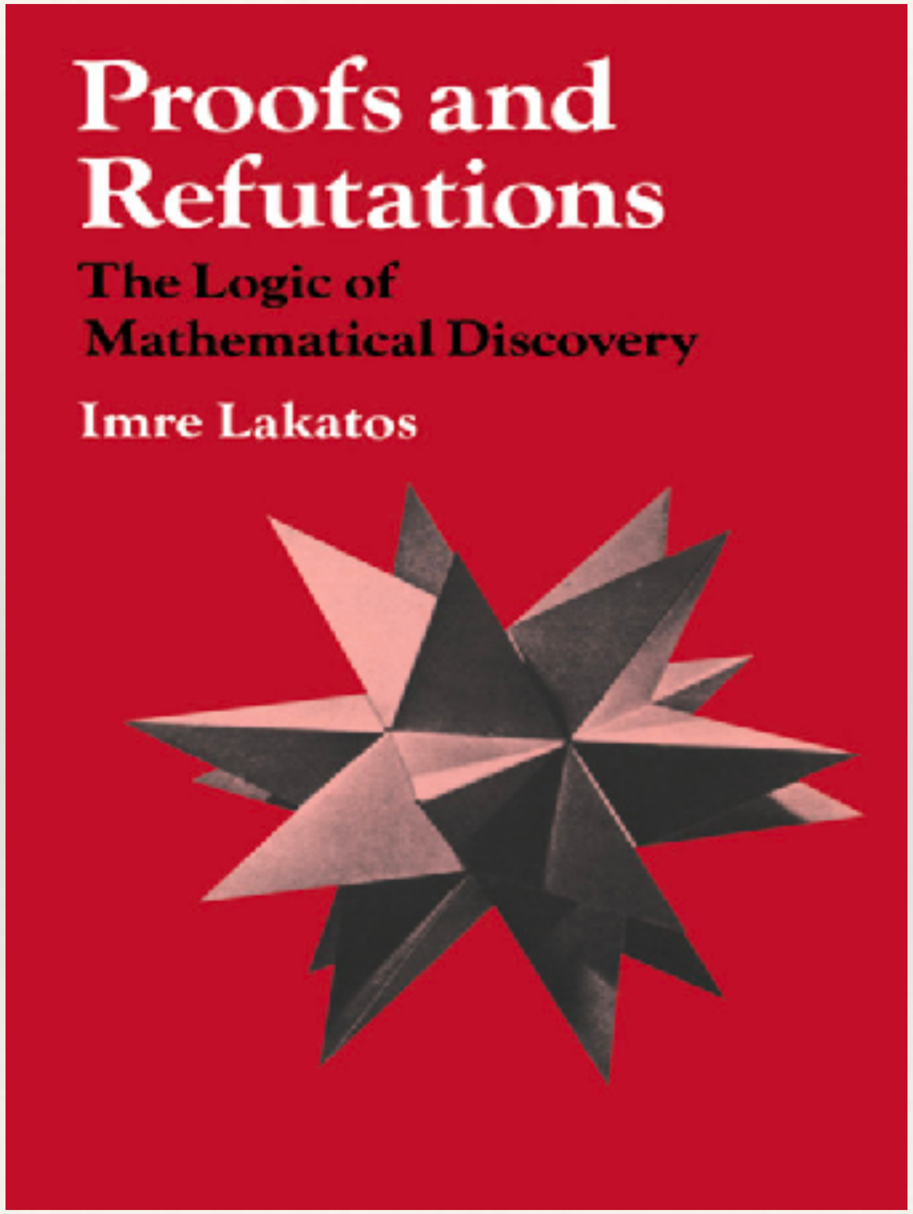
   

4 Vertices, 6 Edges, 4 Faces

Or

1 0-blob (connected component),
0 1-blobs (2-d components)
1 2-blob (3-d components)

Euler’s Formula: V - E + F



Generalization

- Euler Characteristic:
  Alternating sum of vertices, faces, etc.
  Alternating sum of **Betti numbers**
- Betti numbers:
  Number of "holes" at each dimension
  Rank of the n-th **homology group**
- Homology group:
  Group constructed from dissecting an object into n-blobs and finding the cycles
  Function on adjacent components of a **chain complex**

Formal definiti

A **chain complex** $(A_\bullet$ … alled **boundary operators** or **differentials**) $d_n : A_n \to$ … $=0_{n-1}$—the identity in A_{n-1}—for all a in A_{n+1}) for all n. The …

$$\cdots \to A_{n+1} \xrightarrow{d_n} \cdots$$

~~Homological Computations~~ for Term Rewriting Systems

Monoids

A **Set**
equipped with a **Binary Operation** and **Distinguished Element**
such that **the operation is associative** and **the element is identity**

Examples:

{T,F} (and, T)

{T,F} (or, F)

{0,1,2…} (+,0)

{1,2,3…} (*,1)

Monoid Presentations

- Motivation: Finite presentation of infinite structure.
- All monoids are quotients of free monoids.
- A **Set**
  Another **Set**, consisting of pairs of **Words** from the first set.
- Examples:
  {a | _ } (natural numbers under addition)
  {a | aa = a} (the boolean lattice)
  {p,q | pq = 1} (the bicyclic monoid)
  {a,b | aa = a, bb = b} (the free band on two elements)
- All presentations give rise to monoids
  Monoids admit multiple presentations

Monoid Presentations <=> String Rewriting Systems

"The Word Problem"
Given a monoid presentation, find an algorithm to test if two elements are equal under the given rewrite rules.

Emil Post (1947): There are monoids for which equality is undecidable

Proof: Consider a monoid presented by S, K, I. Then look up the "halting problem" on Wikipedia.

Aside: String Rewriting and Computer Science

- Fundamental results in computability
- Instruction sequences in assembly
- Unrestricted grammars
- Combinatory logic
- Operational Transformation
  (edit sequences to documents)
- Distributed and asynchronous systems

A Partial Solution

Knuth/Bendix

Start with a finitely presented monoid.

Create a confluent, normalizing, directed rewrite system (i.e. a *different presentation*).

We do this by systematically *rewriting the rewrite rules*.

It either succeeds, or fails to terminate.

(Newman's lemma: if all **critical pairs** are confluent, the system is globally confluent)

Knuth/Bendix Example

$\{x,y \mid x^3=y^3=(xy)^3=1\}$

1. Create directed reductions in e.g. lexiographic order
   $x^3 \to 1$, $y^3 \to 1$, $(xy)^3 \to 1$

2. Check overlaps to find a **critical pair** (nonconfluent branch)
   $x^3yxyxy \to yxyxy$
   $x^3yxyxy \to x^2$

3. Add a new rule to complete the pair
   $yxyxy \to x^2$

4. Remove rules now made redundant, goto 2.

Result: $x^3 \to 1$, $y^3 \to 1$, $yxyx \to x^2y^2$, $y^2x^2 \to xyxy$

Next Question

- What if we restrict ourselves to finitely presented monoids ***with*** decidable word problems. Can we get a normalization procedure?
- Consider {s,t | sts = tst}
  No normalization is possible.
- But, create a *new presentation* where a=st, and we get.
  {s,t,a | ta->as, st->a, sas->aa, saa->aat}
- So we must establish this as a question over all possible generators.

Moving Between Presentations

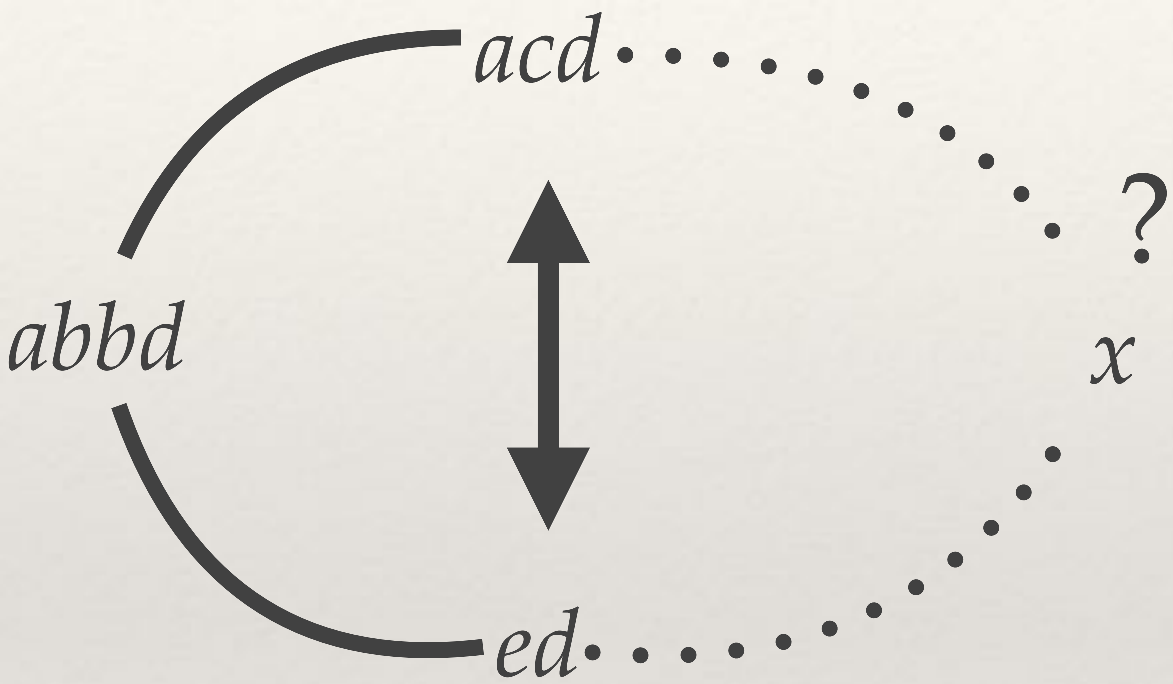
Tietze Transformations:

Add a generator expressed as other generators
Remove a generator expressible by other generators
Add a derivable relation
Remove a redundant relation

The big a-ha

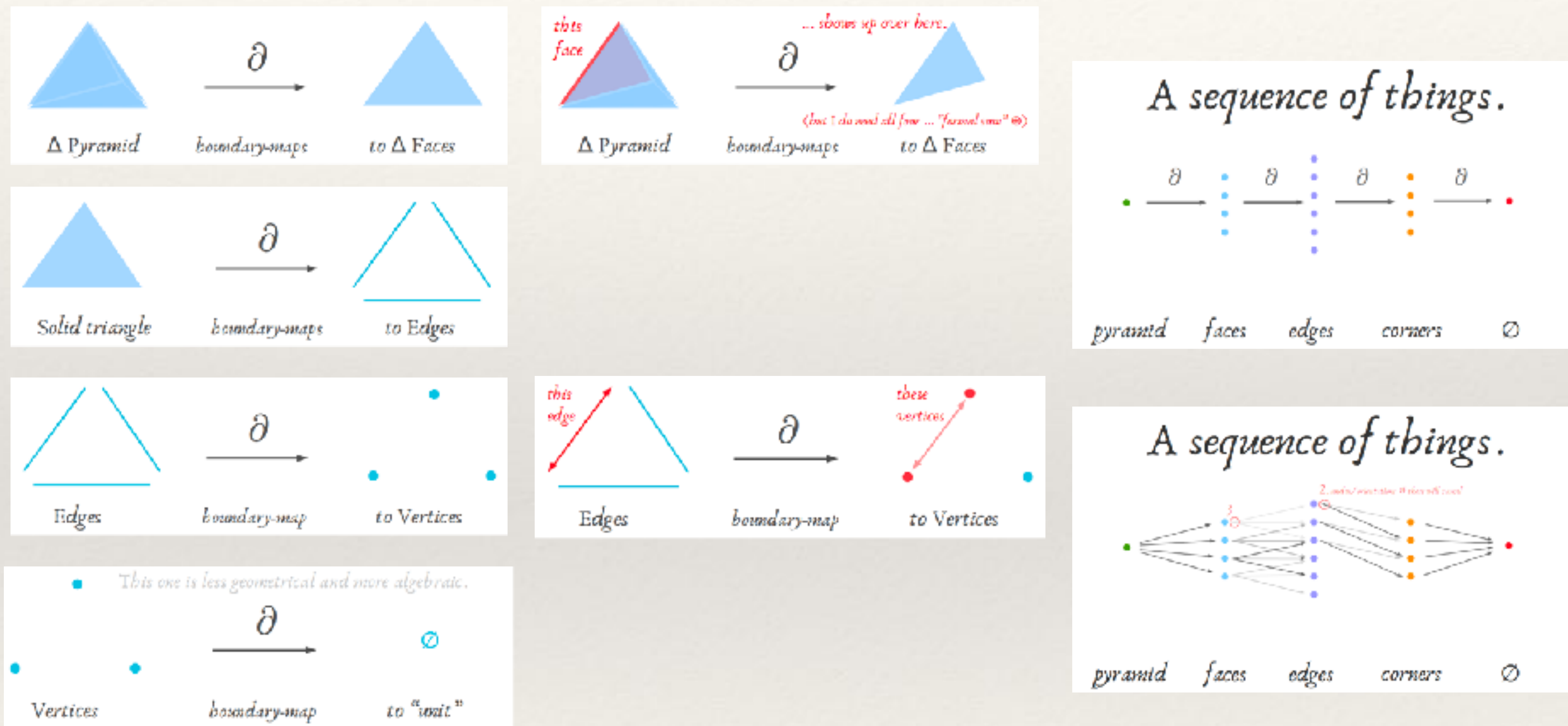
Add a generator <-> add a vertex
Remove a generator <-> delete a vertex
Add a derivable relation <-> add an edge
Remove a redundant relation <-> delete an edge

Rewrite Systems as *Spaces*



Confluence requires a *topological property*: all cycles of a certain shape can be "filled" by a 2-cell.
Find a *homological invariant* of a monoid that is preserved under Tietze transformations.

Chain Complexes Revisited



The **chain condition**: δ^2 = 0.

Our slogan: "The boundary of the boundary is zero"

(source: http://visualizingmath.tumblr.com/post/128146041831/isomorphismes-homology-for-normal-humans-my)

Exact Sequences

Given a chain complex ($A_\bullet$, $d_\bullet$)

$$\cdots \to A_{n+1} \xrightarrow{d_{n+1}} A_n \xrightarrow{d_n} A_{n-1} \xrightarrow{d_{n-1}} A_{n-2} \to \cdots \xrightarrow{d_2} A_1 \xrightarrow{d_1} A_0 \xrightarrow{d_0} A_{-1} \xrightarrow{d_{-1}} A_{-2} \xrightarrow{d_{-2}} \cdots$$

Homology is $\ker(d_n)/\mathrm{im}(d_{n+1})$

Suppose: $\mathrm{im}(d_{n+1}) = \ker(d_n)$. Then the homology is trivial. (no holes), and we are *exact* at n.

Exact sequence: chain such that it is exact at every n.

Resolutions

If we only care about homotopy (or homology) structure, then we want to treat any two spaces with the same associated groups as equivalent. A **weak equivalence** is a map between spaces that introduces an isomorphism on homotopy structure.

A **resolution** of a space is a *weakly equivalent* space subject to some condition (depending on the resolution). It gives a way of "rearranging" a space to make it more understandable.

Homology Resolutions

A plain object (group, module, ring, etc) A, considered as a node in a chain complex yields:
0 -> A -> 0

A *resolution* of A is a new chain complex that shares topological structure. A left resolution, for example, looks like:
… A_2 -> A_1 -> A -> 0

As such, a resolution is an *exact sequence* containing A.

Theorem (Squier 1987)

- We take $\mathbb{Z}M$ as the free ring generated by a monoid M; i.e. polynomials in elements of M. Taking M to have elements {a,b,c} we get:
  5a+2b-3c, 2a-1b+4b, …
- A free $\mathbb{Z}M$-module over a set S, written $\mathbb{Z}M[S]$ contains formal sums of pairs from M and S; i.e. polynomials in pairs from M and S.
  Taking S to have elements {x,y,z} we get:
  2ax + 4cy, ay - az, …

Theorem (Squier 1987)

- Given a presentation (Σ_1,Σ_2) of a M, there is an exact sequence of free $\mathbb{Z}$M-modules:

$$\mathbb{Z}\mathbf{M}[\Sigma_2] \xrightarrow{d_2} \mathbb{Z}\mathbf{M}[\Sigma_1] \xrightarrow{d_1} \mathbb{Z}\mathbf{M} \xrightarrow{\varepsilon} \mathbb{Z} \longrightarrow 0.$$

$$\begin{array}{rcl} \mathbb{Z}\mathbf{M} & \xrightarrow{\varepsilon} & \mathbb{Z} \\ u & \longmapsto & 1 \end{array} \qquad \begin{array}{rcl} \mathbb{Z}\mathbf{M}[\Sigma_1] & \xrightarrow{d_1} & \mathbb{Z}\mathbf{M} \\ [x] & \longmapsto & \overline{x} - 1 \end{array} \qquad \begin{array}{rcl} \mathbb{Z}\mathbf{M}[\Sigma_2] & \xrightarrow{d_2} & \mathbb{Z}\mathbf{M}[\Sigma_1] \\ [\alpha] & \longmapsto & [s(\alpha)] - [t(\alpha)] \end{array}$$

(the overbar is the element of the monoid corresponding to a given generator)

(images: GM16)

Theorem (Squier 1987)

- Given a finite presentation (Σ_1,Σ_2) of a M, there is an exact sequence of free $\mathbb{Z}$M-modules:

$$\mathbb{Z}\mathbf{M}[\Sigma_2] \xrightarrow{d_2} \mathbb{Z}\mathbf{M}[\Sigma_1] \xrightarrow{d_1} \mathbb{Z}\mathbf{M} \xrightarrow{\varepsilon} \mathbb{Z} \longrightarrow 0.$$

$$\begin{array}{rcl} \mathbb{Z}\mathbf{M} & \xrightarrow{\varepsilon} & \mathbb{Z} \\ u & \longmapsto & 1 \end{array} \qquad \begin{array}{rcl} \mathbb{Z}\mathbf{M}[\Sigma_1] & \xrightarrow{d_1} & \mathbb{Z}\mathbf{M} \\ [x] & \longmapsto & \overline{x} - 1 \end{array} \qquad \begin{array}{rcl} \mathbb{Z}\mathbf{M}[\Sigma_2] & \xrightarrow{d_2} & \mathbb{Z}\mathbf{M}[\Sigma_1] \\ [\alpha] & \longmapsto & [s(\alpha)] - [t(\alpha)] \end{array}$$

(the overbar is the element of the monoid corresponding to a given generator)

- Theorem: This is a partial free resolution of length 2, composed of **f**initely generated, **p**rojective modules.

- Hence we say M is of homological type left-FP_2

(images: GM16)

Aside: the bracket

$$\begin{array}{rcl} \mathbb{Z}\mathbf{M} & \xrightarrow{\varepsilon} & \mathbb{Z} \\ u & \longmapsto & 1 \end{array} \qquad \begin{array}{rcl} \mathbb{Z}\mathbf{M}[\Sigma_1] & \xrightarrow{d_1} & \mathbb{Z}\mathbf{M} \\ [x] & \longmapsto & \overline{x} - 1 \end{array} \qquad \begin{array}{rcl} \mathbb{Z}\mathbf{M}[\Sigma_2] & \xrightarrow{d_2} & \mathbb{Z}\mathbf{M}[\Sigma_1] \\ [\alpha] & \longmapsto & [s(\alpha)] - [t(\alpha)] \end{array}$$

$[x]$ is an element of $\mathbb{Z}M[\Sigma_1]$, $\bar{x}$ an element of $\mathbb{Z}M$

$[\alpha]$ is an element of $[\Sigma_2]$, but $s(\alpha)$ is an element of Σ_1^*, not Σ_1 !

So, using a "pun" we define [.] of elements of Σ_1^* : Σ_1^* -> $\mathbb{Z}M[\Sigma_1]$

This is an inductive function (in fact, a fold):

$[.]\ 1 = 0$

$[.]\ uv = [u] + \bar{u}[v]$

(images: GM16)

Theorem (Squier 1987)

- If (Σ_1, Σ_2) is confluent, we can generate Σ_3, given by the "fillers" of the critical branches. Then we extend our sequence like so:

$$\mathbb{Z}\mathbf{M}[\Sigma_3] \xrightarrow{d_3} \mathbb{Z}\mathbf{M}[\Sigma_2] \xrightarrow{d_2} \mathbb{Z}\mathbf{M}[\Sigma_1] \xrightarrow{d_1} \mathbb{Z}\mathbf{M} \xrightarrow{\varepsilon} \mathbb{Z} \longrightarrow 0$$

$$d_3[\gamma] = [s_2(\gamma)] - [t_2(\gamma)]$$

- Theorem: This is a partial free resolution of length 3
- Hence we say M is of homological type left-FP_3

(images: GM16)

Theorem (Squier 1987)

- Every **monoid** is of type left-FP_0
- Every **finitely generated monoid** is of type left-FP_1
- Every **finitely presented monoid** is of type left-FP_2
- Every **finite convergent monoid** is of type left-FP_3

Example (Squier 1987)

Example 4.5. For each non-negative integer k, let S_k denote the monoid defined by the following presentation:

generators: $a, b, t, x_1, \ldots, x_k, y_1, \ldots, y_k$;

relations:
$$\begin{array}{ll} at^n b \to \lambda, & (P_n) \\ x_i a \to atx_i, & (A_i) \\ x_i t \to tx_i, & (T_i) \\ x_i b \to bx_i, & (B_i) \\ x_i y_i \to \lambda. & (Q_i) \end{array}$$

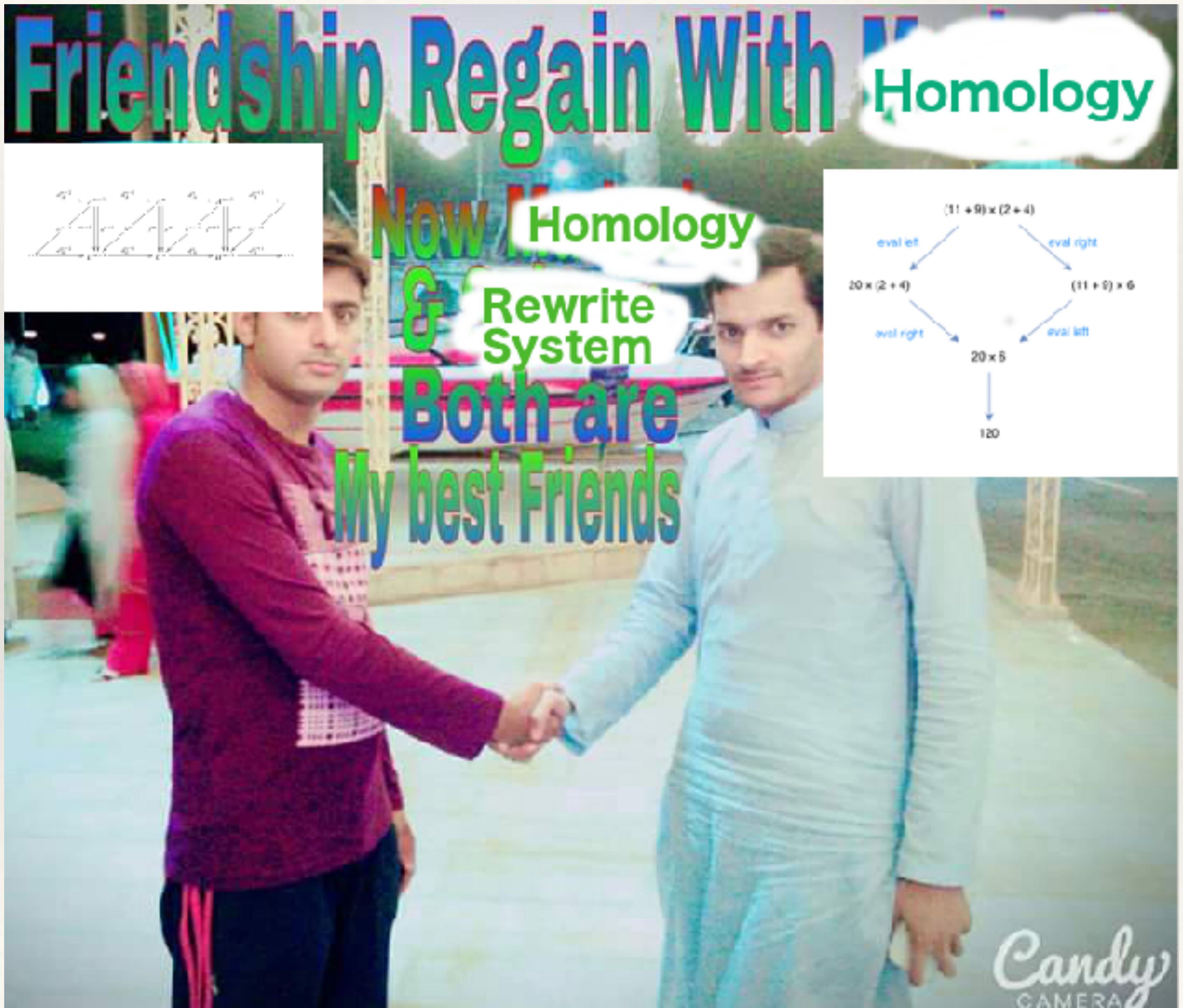
(S_k is proved to have a decidable word problem for all k)

Claim. *If* $k \geq 2$, *then* S_k *is not* $(\mathrm{FP})_3$.

Claim. *If* $k \geq 2$, *then* S_k *does not have a finite uniquely terminating presentation.*

(image: Squier 1987)

Whew!



Friendship Regain With
Homology
Now
Homology
&
Rewrite
System
Both are
My best Friends
(11 + 9) × (2 + 4)
eval left
eval right
20 × (2 + 4)
(11 + 9) × 6
eval right
eval left
20 × 6
120
Candy
CAMERA

Meanwhile in 1987

Cohomology of Algebraic Theories

MAMUKA JIBLADZE AND TEIMURAZ PIRASHVILI

Mathematical Institute, Academy of Sciences,
Rukhadze str. 1, Tbilisi, Georgia 380093, USSR

Communicated by Saunders Mac Lane

Received September 27, 1987

1. INTRODUCTION

Cohomology theory for associative algebras over a field is due to Hochschild [9]. Generalization of this theory for associative algebras over a commutative ring K posed considerable complications. Several definitions have been proposed. For example, in Cartan and Eilenberg's monograph [5], the groups $\mathrm{Ext}^*_{R^e}(R, M)$ are named as candidates for cohomology of the K-algebra R with coefficients in the R–R-bimodule M; here $R^e = R \otimes_K R^{op}$ is the enveloping algebra of R. In MacLane's book [14] Hochschild cohomology is defined in the framework of relative homological algebra,

$$\mathrm{Hoch}^*(R; M) = \mathrm{Ext}^*_{R^e,K}(R, M),$$

where the subscript K signifies that only those extensions which split over K are considered. Still another definition was proposed by Shukla [22], whose cohomology is denoted $\mathrm{Shukla}^*(R; M)$. All these cohomologies are connected by natural homomorphisms:

$$\mathrm{Hoch}^*(R; M) \to \mathrm{Ext}^*_{R^e}(R; M) \to \mathrm{Shukla}^*(R; M).$$

Meanwhile in 1987

String rewriting systems *present* monoids

Term (tree) rewriting systems *present* **algebraic theories.**

As with monoids, we view these things presentation first, but understanding that different presentations may describe the same mathematical object.

Algebraic Theories

An equational theory involves:

Operations with arities (0-ary constants, 1-ary, binary, etc.)
Universally quantified relations over those operations

Example: groups

generating operations: $e : 0$, $- : 1$, $\bullet : 2$

relations: $\forall$ x. x $\bullet$ e = x, $\forall$ x. e $\bullet$ x = x,
$\forall$ x, y, z. (x $\bullet$ y) $\bullet$ z = x $\bullet$ (y $\bullet$ z)
$\forall$ x. x $\bullet$ -x = e, $\forall$ x. -x $\bullet$ x = e

An algebraic theory is an equivalence class of equational theories.

Aside: Term Rewriting and Computer Science

- Typeclasses and laws as theories
- Typeclasses with functional dependencies as a rewrite system
- Syntax trees under equivalence induced by *eval*
- *eval* itself
  (though note: lambda binders mean a theory is not algebraic)
- Computer algebra
- Theorem proving

30 years later…

Monoids correspond to string rewriting systems.

Algebraic theories correspond to *term rewriting systems*.

If homology of monoids lets us prove facts about string rewriting presentations. Then… homology of algebraic theories lets us prove facts about term rewriting systems?

30 years later…

Groups don’t need five relations. In fact, they only need one! (proven in 1952).

x /
((((x / x) / y) / z) /
 (((x / x) / x) / z))
= y

30 years later…

Groups are *one- based*

Semi-lattices and distributive lattices are not. Normal lattices are.

Boolean algebra? Proven one-based in 2,000, with a single axiom of over 40 million symbols.
(this was later improved)

There is a Homology that determines if a theory is one-based

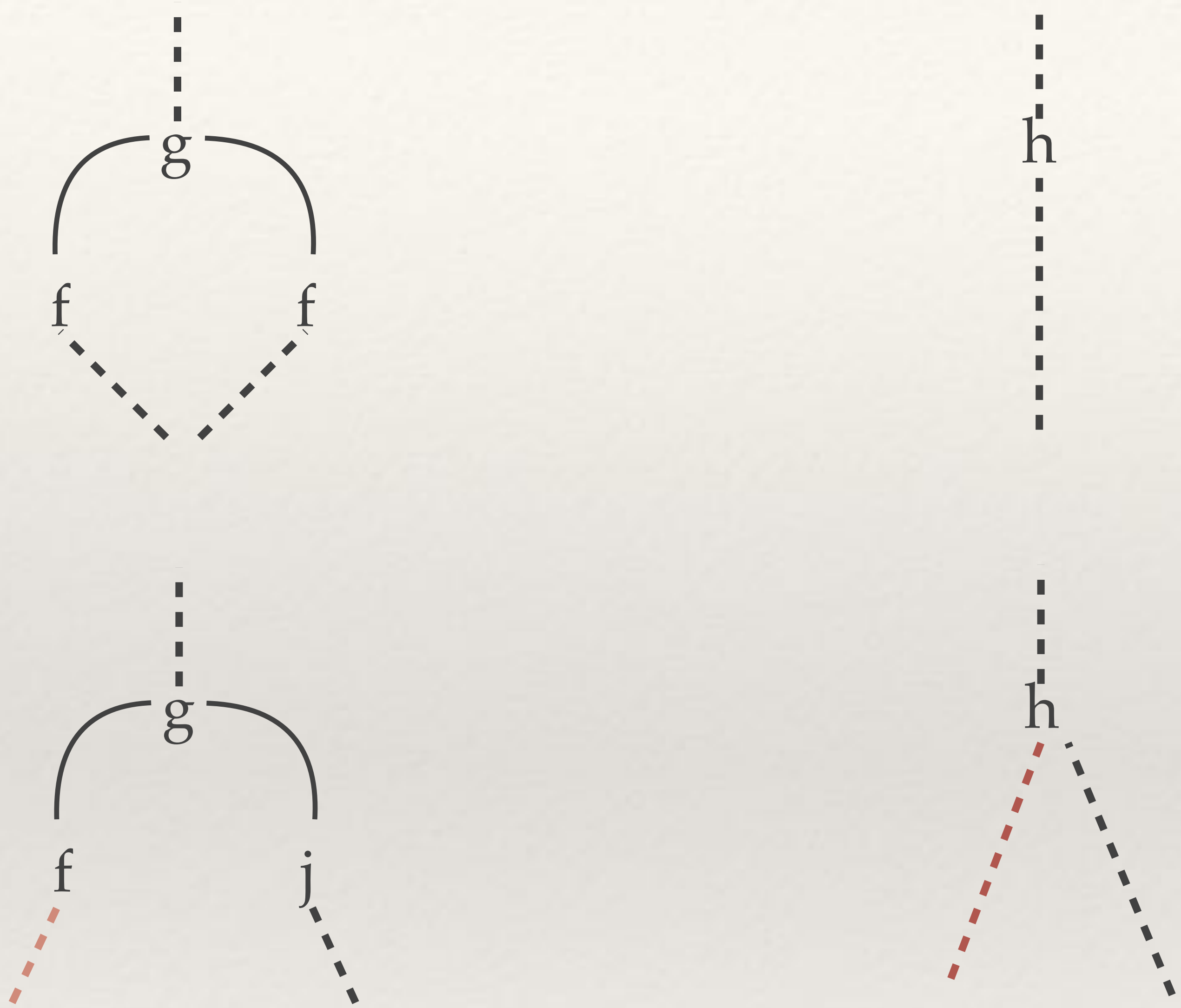
Idea: each rewrite rule *consumes* some symbols, and *produces* other symbols.

We can forget the shape of the rule, and just examine the net effect.

g(f(x),f(x)) = h(x) —> h = 2f + g

however we need to interpret this in a way that is aware of substitutions into contexts.

Aside: Contexts



A **context** in K_n is a term with a distinguished variable and n other variables

A **bicontext** in $\mathbb{K}(m,n)$ is a context in n and an arrow from a term in m to a term in n.

Bicontexts induce functions between terms (in fact, *rewriting* functions).

Contexts make Things Complicated

Monoid —> Ringoid

Free monoid —> Quotient of the free ringoid (by context equivalences induced by the relations), aka R.

▶ **Example 15.** Consider the rewriting system with operations and arities $a : 0$, $b : 0$, $f : 1$, $g : 2$, and two rules $A : a \Rightarrow b$ and $B : f(x_1) \Rightarrow g(x_1, x_1)$. The quotient on contexts is generated by $g(\square, x_1) + g(x_1, \square) - f(\square)$.

(images: MM16)

There is a Homology that determines if a theory is one-based

Theorem: Every convergent presentation of an algebraic theory gives rise to a partial resolution of the form:

$$\mathcal{R}\underline{P_3} \xrightarrow{\partial_2} \mathcal{R}\underline{P_2} \xrightarrow{\partial_1} \mathcal{R}\underline{P_1} \xrightarrow{\partial_0} \mathcal{R}\underline{P_0} \xrightarrow{\varepsilon} \mathcal{Z} \longrightarrow 0$$

with P_1 the generators, P_2 the relations, and P_3 the critical pairs.

($\mathcal{Z}$ here is the trivial R module)

(images: MM16)

There is a Homology that determines if a theory is one-based

$$\mathcal{R}\underline{P_3} \xrightarrow{\partial_2} \mathcal{R}\underline{P_2} \xrightarrow{\partial_1} \mathcal{R}\underline{P_1} \xrightarrow{\partial_0} \mathcal{R}\underline{P_0} \xrightarrow{\varepsilon} \mathcal{Z} \longrightarrow 0$$

This is an exact sequence, so the homology is trivial.
Hence we take homology over this *tensored* by $\mathcal{Z}^{op}$.
(conceptually, this "cancels" the coefficients in R).

Theorem: The rank of H_1 ($= \ker(\mathcal{Z}^{op} \otimes d_0) / \mathrm{im}(\mathcal{Z}^{op} \otimes d_1)$) is a lower bound on the number of operations of a theory.

Theorem: The rank of H_2 ($= \ker(\mathcal{Z}^{op} \otimes d_1) / \mathrm{im}(\mathcal{Z}^{op} \otimes d_2)$) is a lower bound on the number of relations of a theory.

(images: MM16)

The Homotopification of Everything

"A cardinal principle of modern mathematical research may be stated as a maxim: 'One must always topologize'"

–Marshall Stone (1938)

The Homotopification of Everything

"But fundamental psychological changes also occur… Instead of sets, clouds of discrete elements, we envisage some sorts of vague spaces, which can be very severely deformed, mapped one to another, and all the while the specific space is not important, but only the space up to deformation. If we really want to return to discrete objects, we see continuous components, the pieces whose form or even dimension does not matter. Earlier, all these spaces were thought of as Cantor sets with topology, their maps were Cantor maps, some of them were homotopies that should have been factored out, and so on….

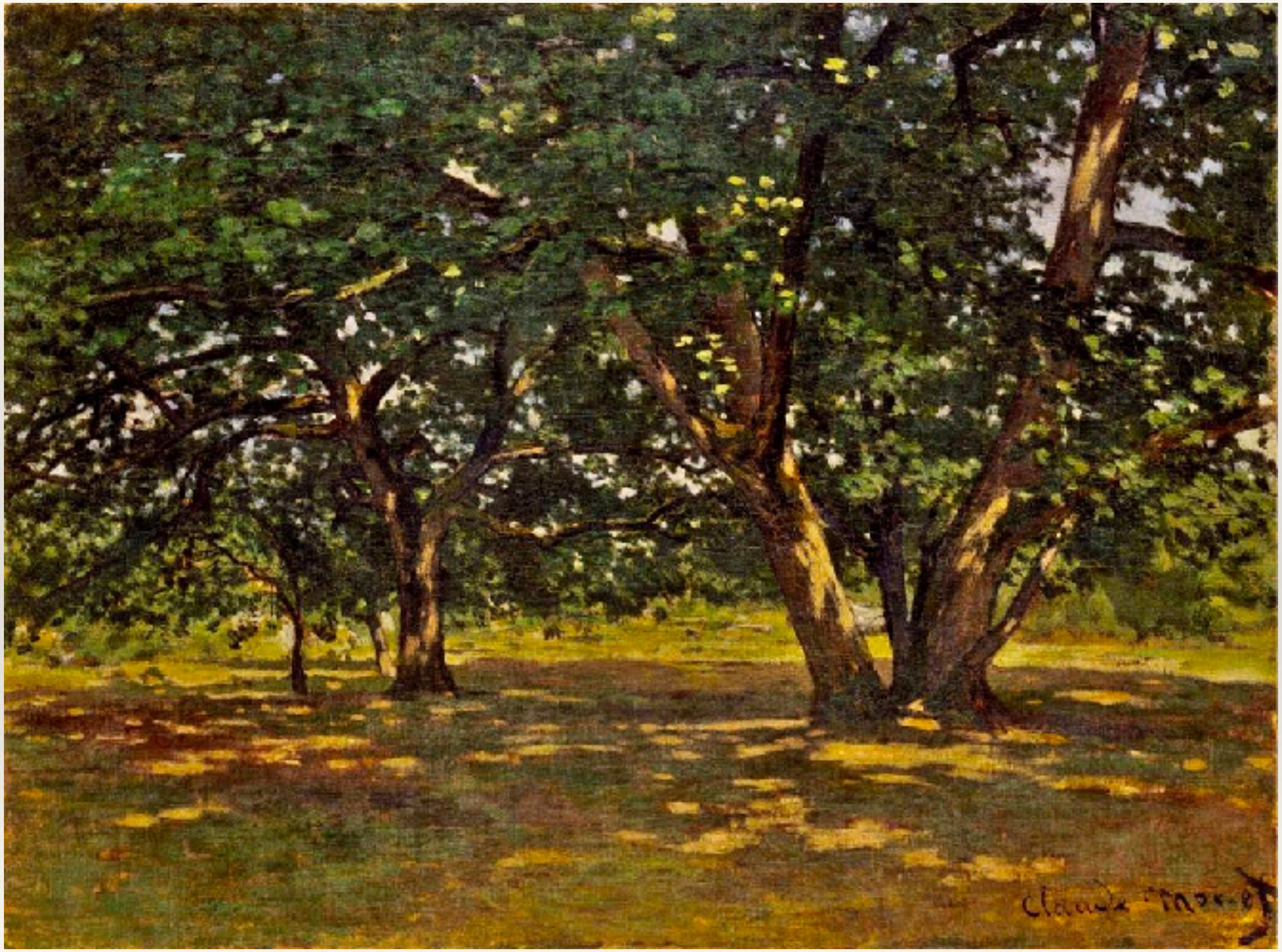
–We Do Not Choose Mathematics as OurProfession, It Chooses Us: Interview with Yuri Manin (2009)

The Homotopification of Everything

"I am pretty strongly convinced that there is an ongoing reversal in the collective consciousness of mathematicians: the right hemispherical and homotopical picture of the world becomes the basic intuition, and if you want to get a discrete set, then you pass to the set of connected components of a space defined only up to homotopy. That is, the Cantor points become continuous components, or attractors, and so on — almost from the start. Cantor's problems of the infinite recede to the background: from the very start, our images are so infinite that if you want to make something finite out of them, you must divide them by another infinity."

–*We Do Not Choose Mathematics as OurProfession, It Chooses Us: Interview with Yuri Manin (2009)*

The Tree and the Shadows



(*Fontainebleau Forest*, Monet, 1865)

References

- More on Squier's Theorem:
  Polygraphs of Finite Derivation Type
  (Giuraud, Malbos, 2016) [GM16]
  Word Problems and a Homological Finiteness Condition for Monoids
  (Squier, 1987)
- More on Algebraic Topology:
  Algebraic Topology (Hatcher, 2002).
- More on Homological Algebra:
  Introduction to Commutative Algebra (Atiyah, MacDonald, 1969).
- More on Groups:
  Group Theory (Course notes by J.S. Milne, 1996 onwards).

(All otherwise unattributed mathematical images sourced from Wikimedia Commons)
(Memes due to Asif Raza Rana)