(Un)expected Encounters of Algebraic Complexity Theory in Machine Learning

Markus Bläser

Saarland University

WACT 2025, Bochum

Probabilistic Circuits

Identification in Structural Causal Models


I would like to discuss lower bounds for syntactically multi-linear circuits...


I would like to discuss lower bounds for syntactically multi-linear circuits...
I know everthing about it...

Probabilistic Modelling

Setup:

- Random variables $X_1, X_2, \dots$ with a joint probability distribution p.
- State space $\mathrm{val}(X_1) \times \mathrm{val}(X_2) \times \dots$.

Evidence query:

- Given $\xi_i \in \mathrm{val}(X_i)$
- compute $\Pr[X_1 = \xi_1, X_2 = \xi_2, \dots]$

Efficiently possible if we can evaluate p efficiently.

Marginal queries

Example:

- $X_C = \begin{cases} 1 & \text{if the train station in city C is closed} \\ 0 & \text{otherwise} \end{cases}$
- joint probability distribution on $\mathrm{Date} \times \bigtimes_{\text{cities C}} X_C$
- What is $\Pr[\mathrm{Date} = \mathrm{Mar}\ 31, X_{\mathrm{Bochum}} = 1]$?

Marginal queries

Example:

- $X_C = \begin{cases} 1 & \text{if the train station in city } C \text{ is closed} \\ 0 & \text{otherwise} \end{cases}$
- joint probability distribution on $\mathrm{Date} \times \bigtimes_{\text{cities } C} X_C$
- What is $\Pr[\mathrm{Date} = \mathrm{Mar}\ 31, X_{\mathrm{Bochum}} = 1]$?
- We need to "marginalize out" all other cities $\longrightarrow$ exponential sum

Marginal queries

Example:

- $X_C = \begin{cases} 1 & \text{if the train station in city } C \text{ is closed} \\ 0 & \text{otherwise} \end{cases}$
- joint probability distribution on $\mathrm{Date} \times \bigtimes_{\text{cities } C} X_C$
- What is $\Pr[\mathrm{Date} = \mathrm{Mar}\ 31, X_{\mathrm{Bochum}} = 1]$?
- We need to "marginalize out" all other cities $\longrightarrow$ exponential sum
- Hardness of marginal queries supported by practical evidence

Probabilistic Circuits

Probabilistic circuits (PC)

- compute probablity mass functions
- input nodes are “basic” probability distributions
  - Gaussian distributions (we do not get polynomials)
  - Categorial distributions: $\Pr[X_i = 1]x_i + (1 - \Pr[X_i = 1]\bar{x}_i$.
- product nodes
- sum nodes computing weighted sums (with nonnegative weights)

Evidence queries are possible in polynomial times in the circuit size.
What about marginal queries?

Decomposable circuits

Definition

A PC is decomposable if the scopes of the children of each product gate are disjoint.

Theorem

If a distribution is given by a decomposable PC, then we can efficiently perform marginalization.

Probabilistic Generating Circuits

- Given categorical variables $X_1, \ldots, X_n$ with image $\{0, \ldots, d-1\}$
- and joint distribution $p(a_1, \ldots, a_n) = \Pr[X_1 = a_1, \ldots, X_n = a_n]$,
- the *probability generating function* is a formal polynomial in formal variables $z_1, \ldots, z_n$ defined by

$$G(z) = \sum_{j_1=0}^{d-1} \cdots \sum_{j_n=0}^{d-1} p(j_1, \ldots, j_n) z_1^{j_1} \cdots z_n^{j_n}. \tag{1}$$

Definition (Zhang et al. 2021)

A *probabilistic generating circuit (PGC)* for a probability distribution p is an arithmetic circuit that computes G.

- PCs compute probability mass functions
- PGCs store probability distributions as formal objects

Probabilistic Generating Circuits

Theorem (Zhang et al. 2021)

PGCs support efficient marginalization for binary random variables

PGCs subsume

- PCs
- DPPs (using the algorithm of Mahajan & Vinay).

Determinantal Point Processes (DPP)

Definition

A probability distribution p over n binary random variables $X_1, \ldots, X_n$ is an L-ensemble if there exists a (symmetric) positive semidefinite matrix L such that for all $x \in \{0,1\}^n$ is distributed according to $\det L_x$, where L_x is the principal minor with the ith rows and columns chosen if $x_i = 1$.

- L is called the kernel
- If L is psd, then all $\det L_x$ are nonnegative
- Normalizing constant is $\det(L + I)$.
- Allows for efficient marginalization

Our results

Theorem (Agarwal & B., 2024)

Marginalization is #P-hard for PGC if the categories have size $\geqslant 3$.

Theorem (Agarwal & B., 2024)

For every PGC over binary variables, there is an equivalent nonmonotone PC (that computes the probability mass function.)

$$g(x_1, \overline{x_1}, \ldots, x_n, \overline{x_n}) = f\left(\frac{x_1}{\overline{x_1}}, \frac{x_2}{\overline{x_2}}, \ldots, \frac{x_n}{\overline{x_n}}\right) \cdot \prod_{i=1}^{n} \overline{x_i}.$$

Our Results (2)

It is essential how we store the basic distributions:

- $\sum_{i=0}^{d-1} \alpha_i x^i$
- $\sum_{i=0}^{d-1} \alpha_i z_i$

Theorem (Agarwal & B., 2024)

Let C be a nonmonotone PC of size s computing a probability distribution over categorical random variables $X_1, \ldots, X_n$ such that the polynomial P computed by C is set-multilinear with respect to the partition $\{z_{i,0}, \ldots, z_{i,d-1}\}$, $1 \leqslant i \leqslant n$. Let $A_1, \ldots, A_n \subseteq \{0, \ldots, d-1\}$. Then we can compute $\Pr[X_1 \in A_1, \ldots, X_n \in A_n]$ in time $O(s)$.

ML — ACT dictionary

probabilistic circuit	monotone arithmetic circuit
decomposable	syntactically multilinear
sum product network	monotone layered circuit
subcircuit tree	parse tree
circuit polynomial	sum of parse trees representation
…	…

ML — ACT dictionary

probabilistic circuit	monotone arithmetic circuit
decomposable	syntactically multilinear
sum product network	monotone layered circuit
subcircuit tree	parse tree
circuit polynomial	sum of parse trees representation
...	...

Conclusion: We are not as smart as we think

Interesting Questions

- Are there more expressive models that are still tractable?
- Are the “generalizations” of DPPs?
- Are there models that subsume DPPs and PCs and compute a nonnegative function *by design*? Formally: Is there a large circuit class such that given $\mathcal{C}$, we can decide in polynomial time whether $\mathcal{C}$ is nonnegative on the part of the domain of interest?

Probabilistic Circuits

Identification in Structural Causal Models

Polynomial Identity Testing

We all love PIT!

Polynomial Identity Testing

We all love PIT!

We investigate algorithms for the problem of Polynomial Identity testing (PIT). Given a polynomial in some implicit representation, it asks whether the polynomial is identically zero or not. It is a fundamental problem in algorithms and complexity theory. It has found applications in algorithm design, for example in algorithms for perfect matching in graphs [17, 36, 39], for primality testing [2, 3, 4], for equivalence testing of read once branching programs [13], and for multi-set equality testing [14], and also in complexity theory, for example, in establishing some major results related to interactive proofs and probabilistically-checkable proofs [38, 9, 8, 7, 45]. In fact, it has also been discovered that an algorithm for polynomial identity testing is intimately connected with complexity theoretic lower bounds [31, 1].

Causation versus Correlation?

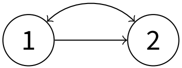
Does smoking cause lung cancer?

- Random variable X_1 = number of cigarettes smoked
- $X_1 = f(\epsilon_1)$ is a function of an unknown random variable ϵ_1.
- Random variable X_2 = binary random variable whether one develops lung cancer.
- $X_2 = g(X_1, \epsilon_1)$ is a function of X_1 and an unknowm random variable ϵ_1.
- Observed correlation between X_1 and X_2.
- Does X_2 cause X_3 or are ϵ_1 and ϵ_2 are correlated.

Causation versus Correlation?

Does smoking cause lung cancer?

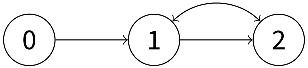
- Random variable X_1 = number of cigarettes smoked
- $X_1 = f(\epsilon_1)$ is a function of an unknown random variable ϵ_1.
- Random variable X_2 = binary random variable whether one develops lung cancer.
- $X_2 = g(X_1, \epsilon_1)$ is a function of X_1 and an unknowm random variable ϵ_1.
- Observed correlation between X_1 and X_2.
- Does X_2 cause X_3 or are ϵ_1 and ϵ_2 are correlated.



Causation versus Correlation?

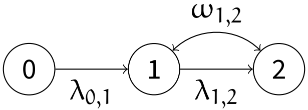
Does smoking cause lung cancer?

- Random variable X_1 = number of cigarettes smoked
- $X_1 = f(\epsilon_1)$ is a function of an unknown random variable ϵ_1.
- Random variable X_2 = binary random variable whether one develops lung cancer.
- $X_2 = g(X_1, \epsilon_1)$ is a function of X_1 and an unknowm random variable ϵ_1.
- Observed correlation between X_1 and X_2.
- Does X_2 cause X_3 or are ϵ_1 and ϵ_2 are correlated.



Identification in Structural Causal Models

- random variables $V_0, V_1, \ldots, V_n$
- $V_i = \sum_j \lambda_{j,i} V_j + \epsilon_i$ with some additional error term ϵ_i
- error terms are normally distributed with zero mean
- covariances between the terms given by $\Omega = (\omega_{i,j})$
- models are *recursive*, i.e., for all $j > i$, we have $\lambda_{j,i} = 0$.



Identification

Given covariances between the V_i, can we recover the $\lambda_{i,j}$ and $\omega_{i,j}$?

In the example:

- $\sigma_{0,1} = \lambda_{0,1}$ and $\sigma_{0,2} = \lambda_{0,1}\lambda_{1,2}$. Thus $\lambda_{1,2} = \frac{\lambda_{0,1}\lambda_{1,2}}{\lambda_{0,1}} = \frac{\sigma_{0,2}}{\sigma_{0,1}}$

Generic Identification

- Given a mixed graph $G = (V, D, B)$, (G, D) is acyclic.
- Relation of observed covariances and parameters is given by

$$\Sigma = (I - \Lambda)^{-1} \Omega (I - \Lambda)^{-T}$$

Definition

G is identifable, if for all Σ (in the image), the parameters Λ (and Ω) are uniquely determined.

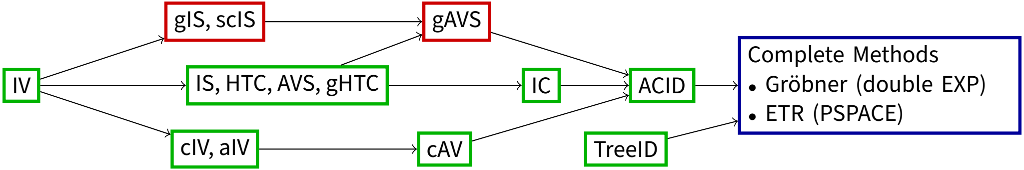
- Solved, but somewhat boring.

Definition

G is generically identifiable, if for almost all Σ (in the image), the parameters Λ (and Ω) are uniquely determined.

Methods for Identification

- Complexity of the problem is widely open



- IV = Instrumental variable
- HTC = Halftrek criterion (Foygel, Draisma, Drton 2012)
- Gröbner basis: Sullivant, Garcia-Puente, Spielvogel 2010
- ETR = Existential theory of the reals (Dörfler et al. 2025)

Hardness of identification

Numerical identification:

- Given a structural causal model and a feasible matrix of observed covariances (Σ).
- Decide whether there is one ore more solutions (Λ and Ω).

Theorem (Dörfler et al. 2024)

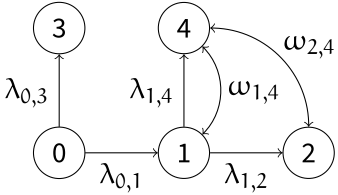
Numerical identification is hard for $\exists\mathbb{R}$.

- $\exists\mathbb{R}$ = existential theory of the reals.
- still hard if we plant an easy solution.

Tree-shaped Structural Causal Models

Definition (Tree-shaped SCM)

If the directed edges form a directed tree with root 0, then the SCM is called tree-shaped.



Key feature: Each node has at most one incoming directed edge.

Theorem (Van der Zander, AISTATS 2022)

Generic identification in tree-shaped SCMs is in PSPACE.*

Our result

Theorem (Gupta & B, 2024)

There is an randomized polynomial algorithm that given a tree-shaped SCM $\mathcal{M}$, determines for each parameter $\lambda_{i,j}$ whether it is

- *generically identifiable or*
- *generically 2-identifiable or*
- *generically unidentifiable.*

In the first two cases, it provides corresponding expressions.

Main features:

- polynomial running time
- completeness for tree-shaped SCMs
- 100% PIT

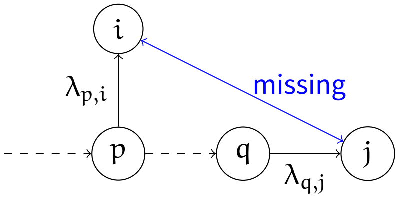
Identification in Tree-shaped SCMs

Theorem (Van der Zander et al., AISTATS 2022)

$\lambda_{x,y}$ *is generically identifiable (*k*-identifiable) iff the system*

$$\begin{aligned} \lambda_{p,i}\lambda_{q,j}\sigma_{p,q} - \lambda_{p,i}\sigma_{p,j} - \lambda_{q,j}\sigma_{i,q} + \sigma_{i,j} &= 0 \quad \text{for each missing } i \leftrightarrow j \\ \lambda_{p,i}\sigma_{0,p} - \sigma_{0,i} &= 0 \quad \text{for each missing } 0 \leftrightarrow i \end{aligned}$$

has a unique solution for $\lambda_{x,y}$ *(*k *solutions, respectively).*



Möbius transforms

- undirected graph G
- nodes are variables
- edges are labeled with a bilinear equation in the incident variables:

$$axy - bx + cy - d = 0$$

- To eliminate y, write

$$y = \frac{bx + d}{ax + c}$$

- This is a Möbius transform. Inverse is given by the adjoint matrix.
- We assume that all $\left(\begin{smallmatrix} b & d \\ a & c \end{smallmatrix}\right)$ have rank two.
- In the application, rank one edges can be dealt separately. (Rank testing = PIT)

Solving systems of bilinear equations

- Let

$$axy - bx + cy - d = 0,$$
$$Ayz - By + Cz - D = 0$$

  be two such bilinear equations, sharing the indeterminate y.
- To eliminate y, write

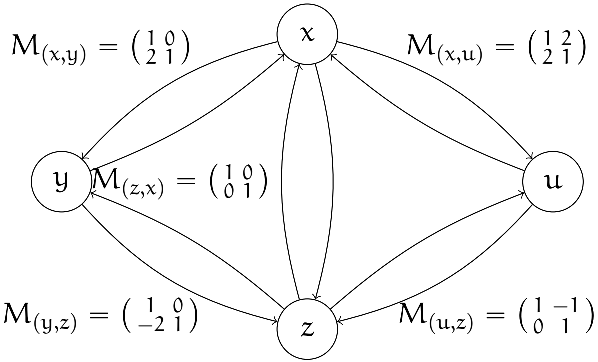
$$y = \frac{bx + d}{ax + c}, \quad z = \frac{By + D}{Ay + C}$$

  and obtain

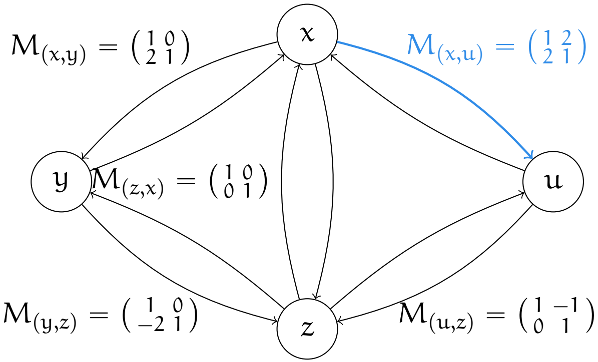
$$z = \frac{(Bb + Da)x + (Bd + Dc)}{(Ab + Ca)x + (Ad + Cc)}.$$

- New coefficients: $\left(\begin{smallmatrix} B & D \\ A & C \end{smallmatrix}\right)\left(\begin{smallmatrix} b & d \\ a & c \end{smallmatrix}\right) = \left(\begin{smallmatrix} Bb+Da & Bd+Dc \\ Ab+Ca & Ad+Cc \end{smallmatrix}\right)$.
- Composition of Möbius transforms

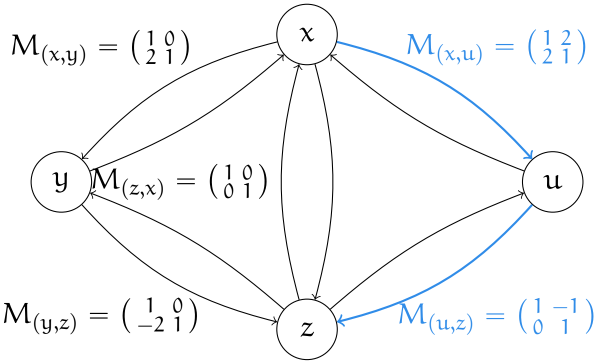
Identification using cycles



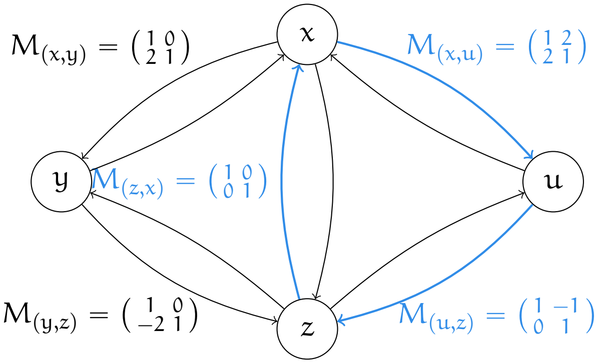
Identification using cycles



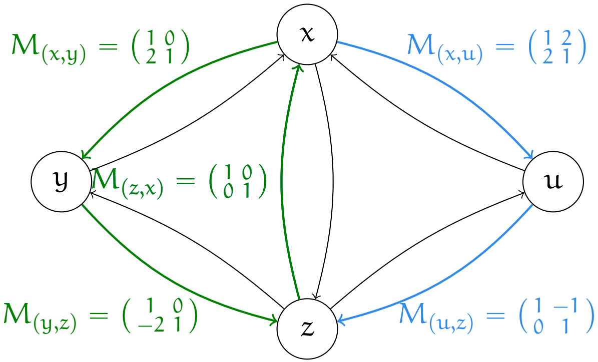
Identification using cycles



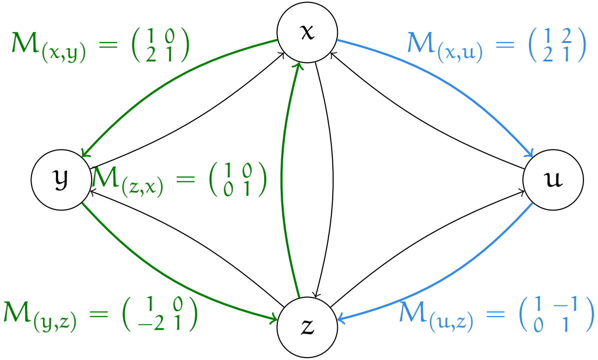
Identification using cycles



Identification using cycles



Identification using cycles



- Two equations for x of the form $x = \frac{bx+d}{ax+c}$, but they could be trivial.
- If one is nontrivial, solution can be propagated to all other variables.

When is an equation trivial?

Lemma

Let $x_1, \dots, x_t, x_1$ be a simple directed cycle and $\left(\begin{smallmatrix} b & d \\ a & c \end{smallmatrix}\right)$ be the corresponding product.

1. *If $a \neq 0$, then x_1 has at most two solutions (depending on the discriminant).*
2. *If $a = 0$ but $c - b \neq 0$, then x_1 has exactly one solution.*
3. *If $a = c - b = 0$ but $d \neq 0$, then there is no solution.*
4. *If $a = c - b = d = 0$, then x_1 has infinitely many solutions.*

1. Solutions given by $\frac{-(c-b)\pm\sqrt{\Delta}}{2a}$. Zero solutions cannot happen here.
2. Solutions given by $\frac{d}{c-b}$ (rationally identifiable).
3. Cannot happen in our application.

$\longrightarrow$ 100% PIT

How to find such an identifying cycle?

- Van der Zander et al. enumerate all cycles (PSPACE).

Definition

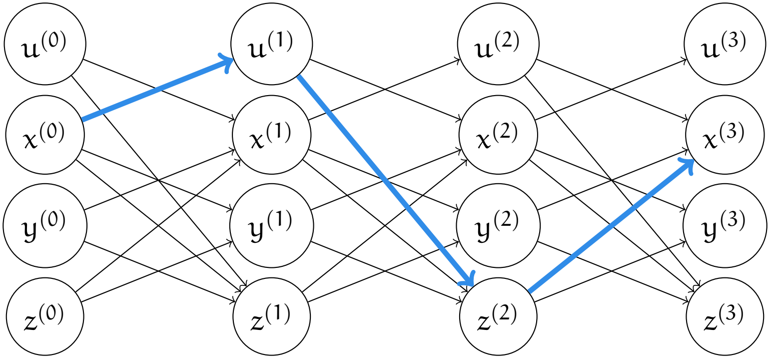
We call a closed walk identifying if $\begin{pmatrix} b & d \\ a & c \end{pmatrix}$ is not a multiple of the identity matrix.

- Let W be the weighted adjacency matrix with 2×2-matrices as entries.
- Diagonal elements of W^t are the sums of all matrices of closed walks of length t.
- Problem 1: We need simple walks.
- Problem 2: Cancellations

Lemma

If there is an identifying walk of length t, then there is an identifying simple walk of length $\leqslant t$.

Finger printing



- weight of $(v_i^{(k)}, v_j^{(k+1)})$ is $x_{i,j}^{(k+1)} w(v_i, v_j)$, $w(v_i, v_j)$ weight of original edge.
- No cancellations. Use PIT to check for an identifying cycle.
- Use self-reduction to find an identifying cycle.

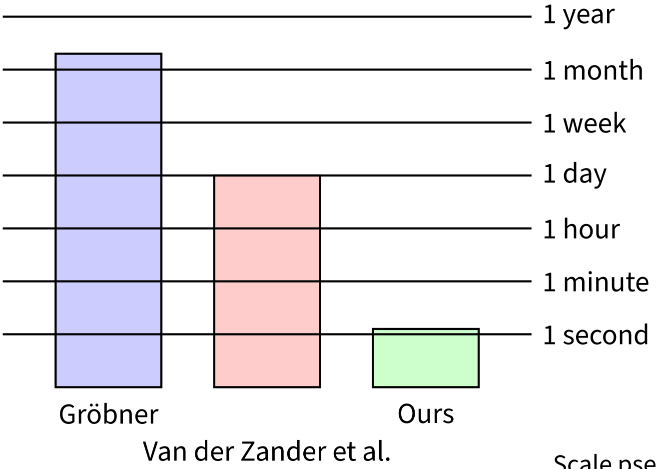
Overall algorithm

Input: A tree-shaped mixed graph $M = (V, D, B)$
Output: For each $\lambda_{p,i}$, we output whether it is generically identifiable, 2-identifiable, or unidentifiable. In the first two cases, we output corresponding FASTPs.

Find all rank-1 edges in the missing edge graph $(V, \bar{B})$.
For each missing rank-1 edge $i \leftrightarrow j$, check which of the parameters $\lambda_{p,i}$ or $\lambda_{q,j}$ we can identify. Mark the node i or j, respectively.
Remove all rank-1 edges from the missing edge graph. Call the resulting graph H. Let $C_1, \ldots, C_t$ be the connected components of H.
for each connected component C_i **do**
    if C_i contains a marked node **then**
        Propagate the result to all unidentified nodes in C_i.
    else
        Find an identifying cycle in C_i (using PIT).
        If no such cycle is found, report that all nodes of C_i are unidentifiable.
        If the cycle produces one solution, then progagate it to all the nodes of C_i.
        If the cycle produces two solutions, then propagate it to all the nodes of C_i.
        Plug the solutions into the equations of C_i and use PIT to check whether all equations are satisfied.

Implementation

We benchmarked our code on 856 SCMs with 8 nodes (from van der Zander et al.)



Scale pseudologarithmic

computational complexity

- Since 2025, I am the editor-in-chief of computational complexity.
- Founded by Joachim von zur Gathen in the spirit of Schönhage and Strassen.
- Transformative journal.
- Editorial board contains 5 Turing award winners, 2 Abacus prize winners, 1 Abel prize winner.
- Quantum is covered, too: Thomas Vidick and Francois Le Gall.
- If mathematicians (and physicists) are to take us seriously, we need to submit to journals, too.

Please consider submitting very good papers of yours!