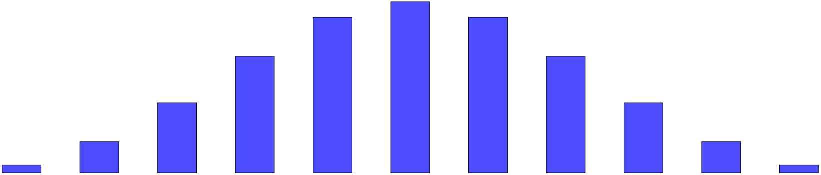
Complexity of Log-concave Inequalities for Matroids

Swee Hong Chan

joint with Igor Pak



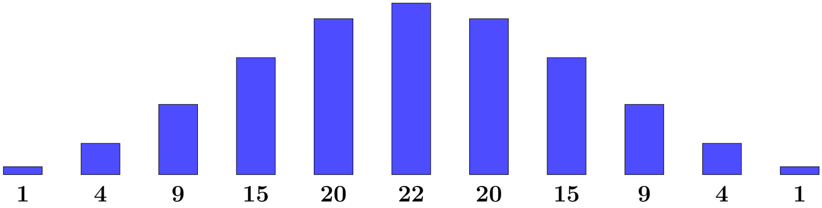
What is log-concavity?

A sequence $a_1, \ldots, a_n \in \mathbb{N}_{\geq 0}$ is log-concave if

$$a_k^2 \;\geq\; a_{k+1}\, a_{k-1} \qquad (1 < k < n).$$

Log-concavity (and positivity) implies unimodality:

$$a_1 \;\leq\; \cdots \;\leq\; a_m \;\geq\; \cdots \;\geq\; a_n \quad \text{for some } 1 \leq m \leq n.$$



Log-concave shaped objects in real life

Cheonmachong (천마총) burial mound,
Gyeongju, South Korea.

Example 1: Binomial coefficients

$$a_k \;=\; \binom{n}{k} \qquad k = 0, 1, \ldots, n.$$

This sequence is log-concave because

$$\frac{a_k^2}{a_{k+1}\,a_{k-1}} \;=\; \frac{\binom{n}{k}^2}{\binom{n}{k+1}\binom{n}{k-1}} \;=\; \left(1+\frac{1}{k}\right)\left(1+\frac{1}{n-k}\right),$$

which is greater than 1.

Example 2: Permutation inversion sequence

Let

$$a_k \; := \; \text{number of } \pi \in S_n \text{ with } k \text{ inversions,}$$

where inversion of π is pair $i < j$ s.t. $\pi_i > \pi_j$.

This sequence is log-concave because

$$\sum_{0 \le k \le \binom{n}{2}} a_k \, q^k \; = \; [n]_q! \; = \; \prod_{i=1}^{n-1} (1 + q + q^2 + \ldots + q^i)$$

is a product of log-concave polynomials.

Example 3: Forests of a graph

a_k = number of forests with k edges of graph G.

Forest is a subset of edges of G that has no cycles.

Log-concavity was conjectured for all matroids (Mason '72), and was proved through combinatorial Hodge theory (Huh '15).

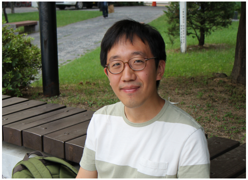
G | forest | not forest | spanning tree

Example 3: Forests of a graph

a_k = number of forests with k edges of graph G.

Forest is a subset of edges of G that has no cycles.

Log-concavity was conjectured for all matroids (Mason '72), and was proved through combinatorial Hodge theory (Huh '15).

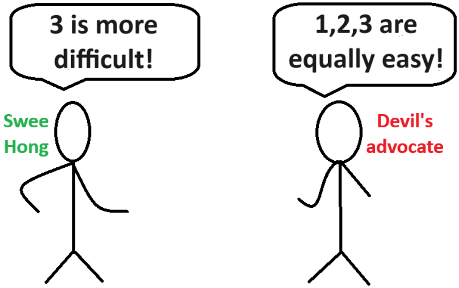


June Huh

Fields Medal

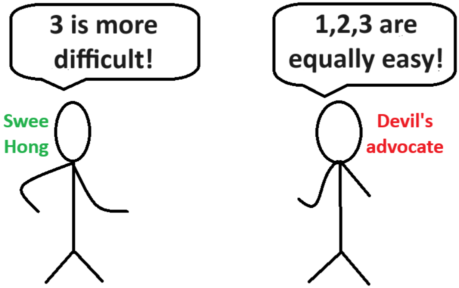
Motivation

Which log-concave inequality is more "difficult"?



Motivation

Which log-concave inequality is more "difficult"?



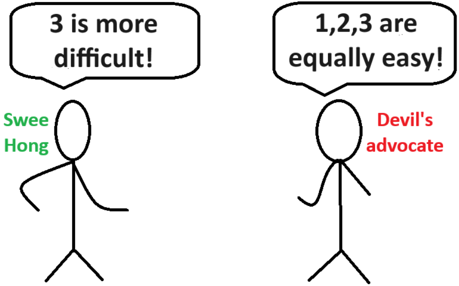
We will show that [redacted] (3)

is **strictly more** difficult than the rest,

using **Complexity Theory**.

Motivation

Which log-concave inequality is more "difficult"?



We will show that a generalization of (3) is **strictly more** difficult than the rest, using **Complexity Theory**.

Matroids

Object: Matroids

Matroid $\mathcal{M} = (X, \mathcal{I})$ is ground set X with collection of independent sets $\mathcal{I} \subseteq 2^X$.

Graphic matroids

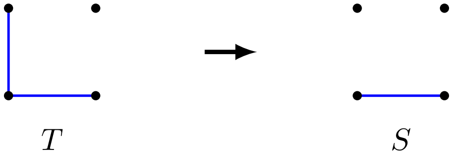
- X = edges of a graph G,
- $\mathcal{I}$ = forests in G.

Binary matroids

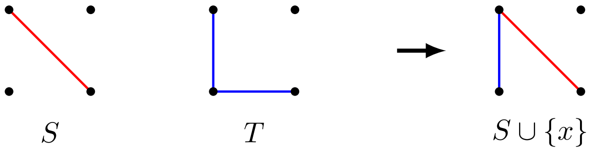
- X = set of vectors over finite field $\mathbb{F}_2$,
- $\mathcal{I}$ = sets of linearly independent vectors.

Matroids: Axioms

- (Hereditary) If $S \subseteq T$ and $T \in \mathcal{I}$, then $S \in \mathcal{I}$.



- (Exchange) If $S, T \in \mathcal{I}$ and $|S| < |T|$, then there is $x \in T \setminus S$ such that $S \cup \{x\} \in \mathcal{I}$.



Matroid: Bases and ranks

A basis of $\mathcal{M}$ is a maximal independent set.

Rank r of $\mathcal{M}$ is the size of the bases.

G Basis 1 Basis 2 Not Basis

Matroid generalizes the notion of vector spaces.

Mason's conjecture

First Mason's conjecture

For matroid $\mathcal{M}$, let

$\mathrm{I}(k) \;:=\;$ no. of independents sets with k elements.

For graphic matroid, $\mathrm{I}(k)$ is no. of forest with k edges.

Conjecture (Mason '72)

The sequence $\mathrm{I}(1), \mathrm{I}(2), \ldots$ *is log-concave,*

$$\mathrm{I}(k)^2 \;\geq\; \mathrm{I}(k+1)\,\mathrm{I}(k-1) \qquad (k \in \mathbb{N}),$$

First Mason's conjecture (continued)

Conjecture (Mason '72)

$$\mathrm{I}(k)^2 \;\geq\; \mathrm{I}(k+1)\,\mathrm{I}(k-1) \qquad (k \in \mathbb{N}).$$

Conjecture was proved for graphic matroids by (Huh '15), and for all matroids by (Adiprasito–Huh–Katz '18).

Both proofs used combinatorial Hodge theory.

First Mason's conjecture (continued)

Conjecture (Mason '72)

$$\mathrm{I}(k)^2 \;\geq\; \mathrm{I}(k+1)\,\mathrm{I}(k-1) \qquad (k \in \mathbb{N}).$$

Conjecture was proved for graphic matroids by (Huh '15), and for all matroids by (Adiprasito–Huh–Katz '18).

Both proofs used combinatorial Hodge theory.

We will show that Mason's conjecture is consequence of a stronger inequality.

Stanley–Yan inequality

Stanley–Yan inequality (simple case)

Let $\mathcal{M}$ be a matroid with ground set X and rank r.

Fix a subset S of X. Let

$\mathrm{B}(k)$:= no. of **bases** B such that $|B \cap S| = k$,

multiplied by $r! \times \binom{r}{k}^{-1}$.

Theorem (Stanley '81, Yan '23)

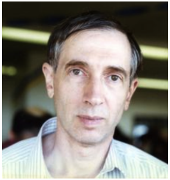
The sequence $\mathrm{B}(1), \mathrm{B}(2), \ldots$ *is log-concave,*

$$\mathrm{B}(k)^2 \ \geq \ \mathrm{B}(k+1)\,\mathrm{B}(k-1) \qquad (k \in \mathbb{N}).$$

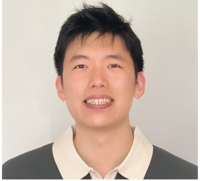
Stanley–Yan inequality (simple)

Theorem (Stanley '81, Yan '23)

$$\mathrm{B}(k)^2 \ \geq \ \mathrm{B}(k+1)\,\mathrm{B}(k-1) \qquad (k \in \mathbb{N}).$$



Richard Stanley



Alan Yan

Stanley–Yan inequality (simple)

Theorem (Stanley '81, Yan '23)

$$\mathrm{B}(k)^2 \ \geq \ \mathrm{B}(k+1)\,\mathrm{B}(k-1) \qquad (k \in \mathbb{N}).$$

Proved for regular matroids by (Stanley '81) using Alexandrov–Fenchel inequality for mixed volumes.

Proved for all matroids by (Yan '23) using theory of Lorentzian polynomials.

Proof of Mason's conjecture using Stanley–Yan inequality

Proof of Mason's conjecture using SY inequality

Let

$\mathcal{M}$:= original matroid in Mason's conjecture;

$\mathcal{F}$:= matroid with r elements and with every subset being independent;

$\mathcal{M}'$:= direct sum of $\mathcal{M}$ and $\mathcal{F}$;

S := ground set of $\mathcal{M}$.

Then

$$\mathrm{I}(k) \text{ for } \mathcal{M} \quad = \quad \frac{1}{r!} \times \mathrm{B}(k) \text{ for } \mathcal{M}'.$$

Proof of Mason's conjecture using SY inequality

Since

$$\mathrm{I}(k) \text{ for } \mathcal{M} \quad = \quad \frac{1}{r!} \times \mathrm{B}(k) \text{ for } \mathcal{M}',$$

we then conclude that

Stanley–Yan inequality for $\mathcal{M}'$
implies Mason's conjecture for $\mathcal{M}$.

Stanley–Yan inequality (full version)

Fix $d \geq 0$, disjoint subsets $S, S_1, \ldots, S_d$ of X, and $\ell_1, \ldots, \ell_d \in \mathbb{N}$.

$\mathrm{B}_d(k) :=$ number of bases B of $\mathcal{M}$ such that $|B \cap S| = k,\ |B \cap S_i| = \ell_i$ for $i \in [d]$,

multiplied by $r! \times \binom{r}{k, \ell_1, \ldots, \ell_d}^{-1}$.

Theorem (Stanley '81, Yan '23)

The sequence $\mathrm{B}_d(1), \mathrm{B}_d(2), \ldots$ *is log-concave,*

$$\mathrm{B}_d(k)^2 \geq \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1) \qquad (k \in \mathbb{N}).$$

What we want to do

Theorem (Stanley '81, Yan '23)

The sequence $\mathrm{B}_d(1), \mathrm{B}_d(2), \ldots$ *is log-concave,*

$$\mathrm{B}_d(k)^2 \;\geq\; \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1) \qquad (k \in \mathbb{N}).$$

Both LHS and RHS of this inequality has combinatorial interpretations.

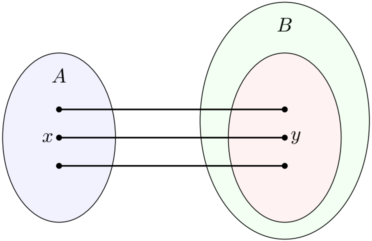
But we will show that this inequality has **no combinatorial injective proof**.

Combinatorial injective proof

Combinatorial injection

An injection $f : A \to B$ is combinatorial if

- Given $x \in A$, the image $f(x)$ is computable in poly($|x|$) steps;
- Given $y \in B$, it takes poly($|y|$) steps to decide if y is in image of f; and if so, the pre-image $f^{-1}(y)$ is computable in poly($|y|$) steps.



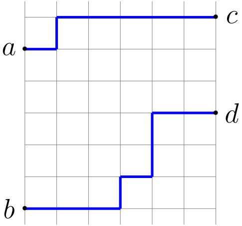
Example: Injective proof of binomial inequality

$$\binom{n}{k}^2 \geq \binom{n}{k+1}\binom{n}{k-1} \qquad (1 < k < n).$$

This inequality has a lattice path interpretation:

$K(a \to c, b \to d) :=$ no. of pairs of north-east lattice paths from a to c and b to d,

for $a, b, c, d \in \mathbb{Z}^2$.



Example: Injective proof of binomial inequality

Let

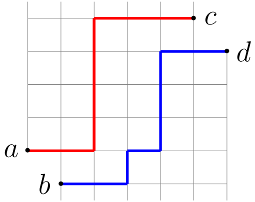
$$a = (0,1), \qquad c = (k, n-k+1),$$

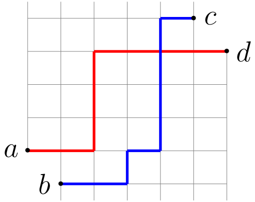
$$b = (1,0), \qquad d = (k+1, n-k).$$

Then

$$K(a \to c, b \to d) = \binom{n}{k}^2,$$

$$K(a \to d, b \to c) = \binom{n}{k-1}\binom{n}{k+1}.$$

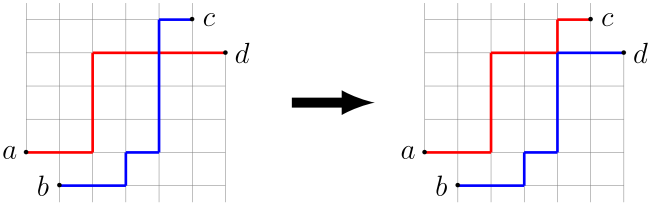




Example: Injective proof of binomial inequality

$$f : K(a \to d, b \to c) \to K(a \to c, b \to d)$$

is defined by path-swapping injections.



Images of f are pairs of lattice paths that intersects.

First main result

Theorem 1 (C.–Pak '24+)

There is **no combinatorial injective proof** *for the Stanley–Yan inequality, assuming polynomial hierarchy does not collapse.*

The assumption above is slightly stronger than $\mathsf{P} \neq \mathsf{NP}$, and is widely used in Complexity Theory.

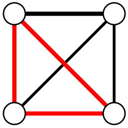
Polynomial hierarchy

Level 0: Complexity class P

P := Decision problems that, given input x, can be solved in poly($|x|$) time.

Example (Problem in P)

Does a graph G contain a triangle?



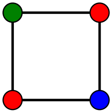
This complexity class is denoted by Σ_0^{P}.

Level 1: Complexity class NP

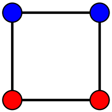
NP := Problems asking about existence of a solution S for input x, where validity of S can be verified in poly($|x|$) time.

Example (Problem in NP)

Does a graph G have a proper 3-coloring?



Proper coloring

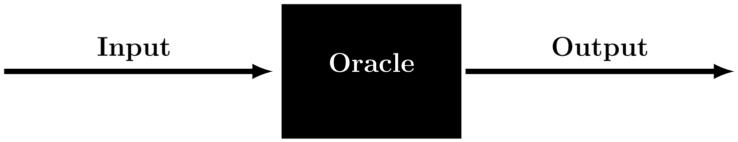


Improper coloring

This complexity class is denoted by Σ_1^P.

Oracle machine

An oracle machine is a black box capable of solving problems from a given class in a single operation.



Level i of polynomial hierarchy

The class $\Sigma_i^{\mathsf{P}} := \mathsf{NP}^{\Sigma_{i-1}^{\mathsf{P}}}$ is

Problems asking about existence of a solution S for input x, where validity of S can be verified in poly($|x|$) time, augmented by $\Sigma_{i-1}^{\mathsf{P}}$-oracle.

Note that

$$\Sigma_0^{\mathsf{P}} \subseteq \Sigma_1^{\mathsf{P}} \subseteq \Sigma_2^{\mathsf{P}} \subseteq \Sigma_3^{\mathsf{P}} \subseteq \cdots .$$

Polynomial hierarchy (PH)

Polynomial hierarchy is the union of all Σ_i^{P}'s,

$$\mathsf{PH} := \bigcup_{i=0}^{\infty} \Sigma_i^{\mathsf{P}}.$$

Conjecture

Polynomial hierarchy does not collapse,

$$\Sigma_0^{\mathsf{P}} \subsetneq \Sigma_1^{\mathsf{P}} \subsetneq \Sigma_2^{\mathsf{P}} \subsetneq \Sigma_3^{\mathsf{P}} \subsetneq \cdots$$

- $\Sigma_0^{\mathsf{P}} \neq \Sigma_1^{\mathsf{P}}$ is equivalent to $\mathsf{P} \neq \mathsf{NP}$.
- $\Sigma_1^{\mathsf{P}} \neq \Sigma_2^{\mathsf{P}}$ is equivalent to $\mathsf{NP} \neq \mathsf{coNP}$.

Back to the main result

Theorem (C.–Pak '24+)

There is **no combinatorial injective proof** *for the Stanley–Yan inequality, assuming* $\Sigma_2^{\mathsf{P}} \neq \Sigma_3^{\mathsf{P}}$.

Proof ideas

Ingredient 1: Study equality conditions

Let SY-Equal be the decision problem:

Input: Binary matroid $\mathcal{M}$, subsets $S, S_1, \ldots, S_d$, integers $k, \ell_1, \ldots, \ell_d$.

Output: YES if $\mathrm{B}_d(k)^2 = \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1)$.

NO if $\mathrm{B}_d(k)^2 > \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1)$.

Understanding complexity of equality conditions is key to showing combinatorial injections do not exist.

Equality conditions vs combinatorial injections

Suppose that combinatorial injection existed:

$$f\,:\; \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1) \quad\longrightarrow\quad \mathrm{B}_d(k)^2.$$

Then, given $y \in \text{RHS}$, it would take poly($|y|$) time to **verify** if y belongs to image of f.

This would imply `SY-Equal` $\in$ `coNP`.

Problem reduces to showing `SY-Equal` $\notin$ `coNP`.

Ingredient 2: Reduce problem to counting bases

Let #Bases be the counting problem:

Input: Binary matroid $\mathcal{M}$.

Output: Number of bases of $\mathcal{M}$.

Lemma (C.–Pak 24+)

There exists a nondeterministic polynomial-time Turing reduction from #Bases *to* SY-Equal *.*

Strategy: show that #Bases is ‘difficult’, then use Lemma to imply SY-Equal is also “difficult”.

Complexity class #P

NP := Problems asking about **existence** of a solution S for input x, where validity of S can be verified in polynomial time.

#P := Problems asking for **number** of solutions S for input x, where validity of S can be verified in polynomial time.

Example (Problem in #P)

Count the number of proper 3-colorings of graph G.

Ingredient 3: Complexity of #Bases

Theorem (Knapp–Noble '24+)
#Bases is #P-complete for binary matroids.

We would like to use the complexity of #Bases to determine the complexity of SY-Equal.

Ingredient 4: Toda's Theorem

Theorem (Toda '91)

Every problem in PH *has a polynomial-time Turing reduction to a problem in* #P*, i.e.*

$$\mathrm{PH} \subseteq \mathrm{P}^{\#\mathrm{P}}.$$

Theorem allows us to connect complexity of decision problems to complexity of counting problems.

Combine all the ingredients

Start with Toda's Theorem:

$$\mathsf{PH} \quad \subseteq \quad \mathsf{P}^{\#\mathsf{P}}.$$

Since #Bases is #P-complete:

$$\mathsf{PH} \quad \subseteq \quad \mathsf{P}^{\texttt{\#Bases}}.$$

Now reduce #Bases to SY-Equal:

$$\mathsf{PH} \quad \subseteq \quad \mathsf{NP}^{\texttt{SY-Equal}}.$$

Combine all the ingredients

Now suppose that combinatorial injection existed.

Then `SY-Equal` belongs to `coNP`:

$$\mathsf{PH} \subseteq \mathsf{NP}^{\texttt{SY-Equal}} \subseteq \mathsf{NP}^{\Sigma_1^{\mathsf{P}}} = \Sigma_2^{\mathsf{P}} .$$

Thus PH would collapse to the second level.

Combine all the ingredients

Now suppose that combinatorial injection existed.

Then `SY-Equal` belongs to `coNP`:

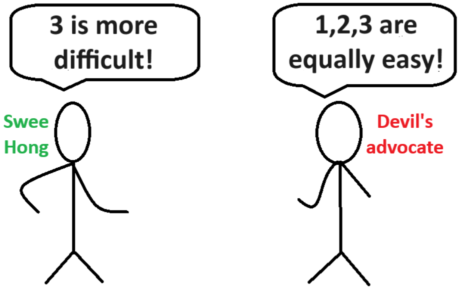
$$\mathsf{PH} \subseteq \mathsf{NP}^{\texttt{SY-Equal}} \subseteq \mathsf{NP}^{\Sigma_1^{\mathsf{P}}} = \Sigma_2^{\mathsf{P}}.$$

Thus PH would collapse to the second level.

Theorem (C.–Pak '24+)

No combinatorial injective proof *for Stanley–Yan ineqquality for binary matroids, assuming* $\Sigma_2^{\mathsf{P}} \neq \Sigma_3^{\mathsf{P}}$.

Recall our goal



We will now show that Stanley–Yan inequality is strictly more difficult than the binomial inequality and permutation inversion inequality.

Second main result

Consider the following computational problem:

Input: Binary matroid $\mathcal{M}$, subsets $S, S_1, \ldots, S_d$,

integers $k, \ell_1, \ldots, \ell_d$.

Output: $\mathrm{B}_d(k)^2 - \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1)$.

Theorem 2 (C.–Pak '24+)

The problem above **does not belong to** #P, *assuming* $\Sigma_2^{\mathsf{P}} \neq \Sigma_3^{\mathsf{P}}$.

Second main result

Theorem (C.–Pak '24+)

The problem of computing

$$\mathrm{B}_d(k)^2 - \mathrm{B}_d(k+1)\,\mathrm{B}_d(k-1)$$

is **not in** #P, *assuming* $\Sigma_2^{\mathsf{P}} \neq \Sigma_3^{\mathsf{P}}$.

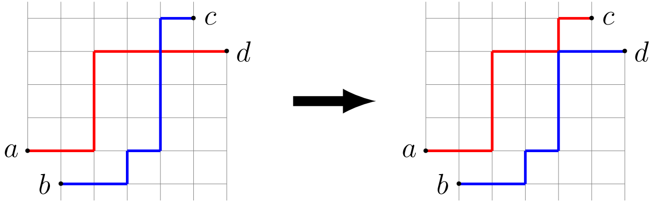
Both LHS and RHS of Stanley–Yan inequality belongs to #P, but their difference does not.

Example 1: Binomial inequality

It follows from path-swapping injections that

$$\binom{n}{k}^2 - \binom{n}{k+1}\binom{n}{k-1} \;=\; \text{number of non-intersecting}$$

lattice paths from a to c and b to d.



Thus the defect of this inequality belongs to #P.

Example 2: Permutation inversion inequality

Let a_k = number of $\pi \in S_n$ with k inversions.

$$\text{Then} \quad \sum_{0 \le k \le \binom{n}{2}} a_k \, q^k \; = \; \prod_{i=1}^{n-1} (1 + q + \ldots + q^i)$$

is computable in poly(n) time.

Thus $a_k^2 - a_{k+1} a_{k-1}$ is computable in poly(n) time;

and thus belongs to #P.

Conclusion

We compare three log-concave inequalities:

Binomial inequality: in #P;

Permutation inversion inequality: in #P;

Stanley–Yan inequality: **not in** #P.

This differentiates **Stanley–Yan inequality** from binomial inequality and permutation inversion inequality.

Open Problem

Conjecture

Defect of Mason's conjecture

$$\mathrm{I}(k)^2 \; - \; \mathrm{I}(k+1)\,\mathrm{I}(k-1) \quad \notin \; \#\mathrm{P}.$$

We have shown defect of Stanley–Yan inequality does not belong to #P, but not Mason's conjecture.

THANK YOU!

Preprint: www.arxiv.org/abs/2407.19608

Webpage: www.math.rutgers.edu/~sc2518/

Email: sweehong.chan@rutgers.edu