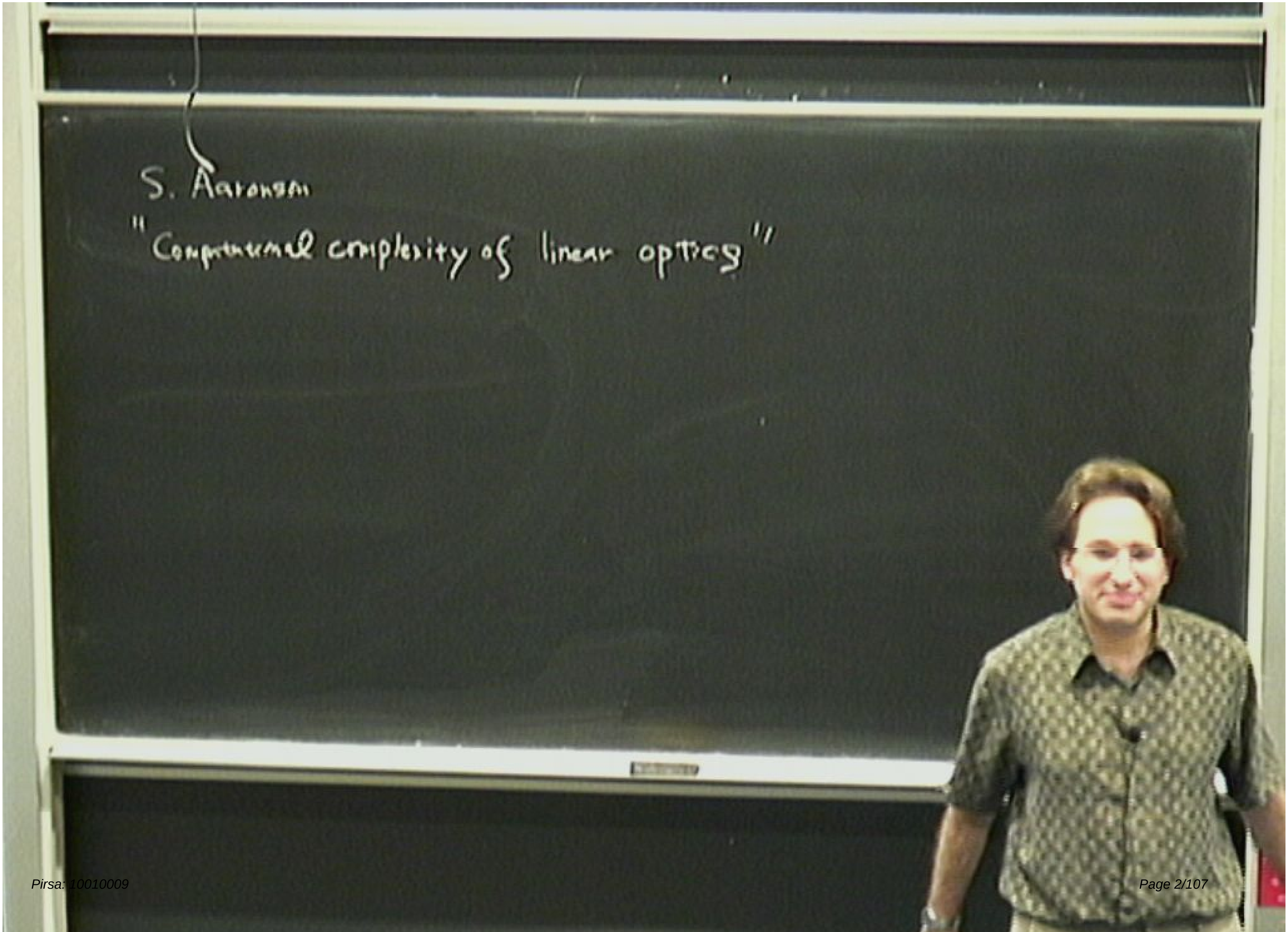
Title: The Computational Complexity of Linear Optics

Date: Jan 25, 2010 04:00 PM

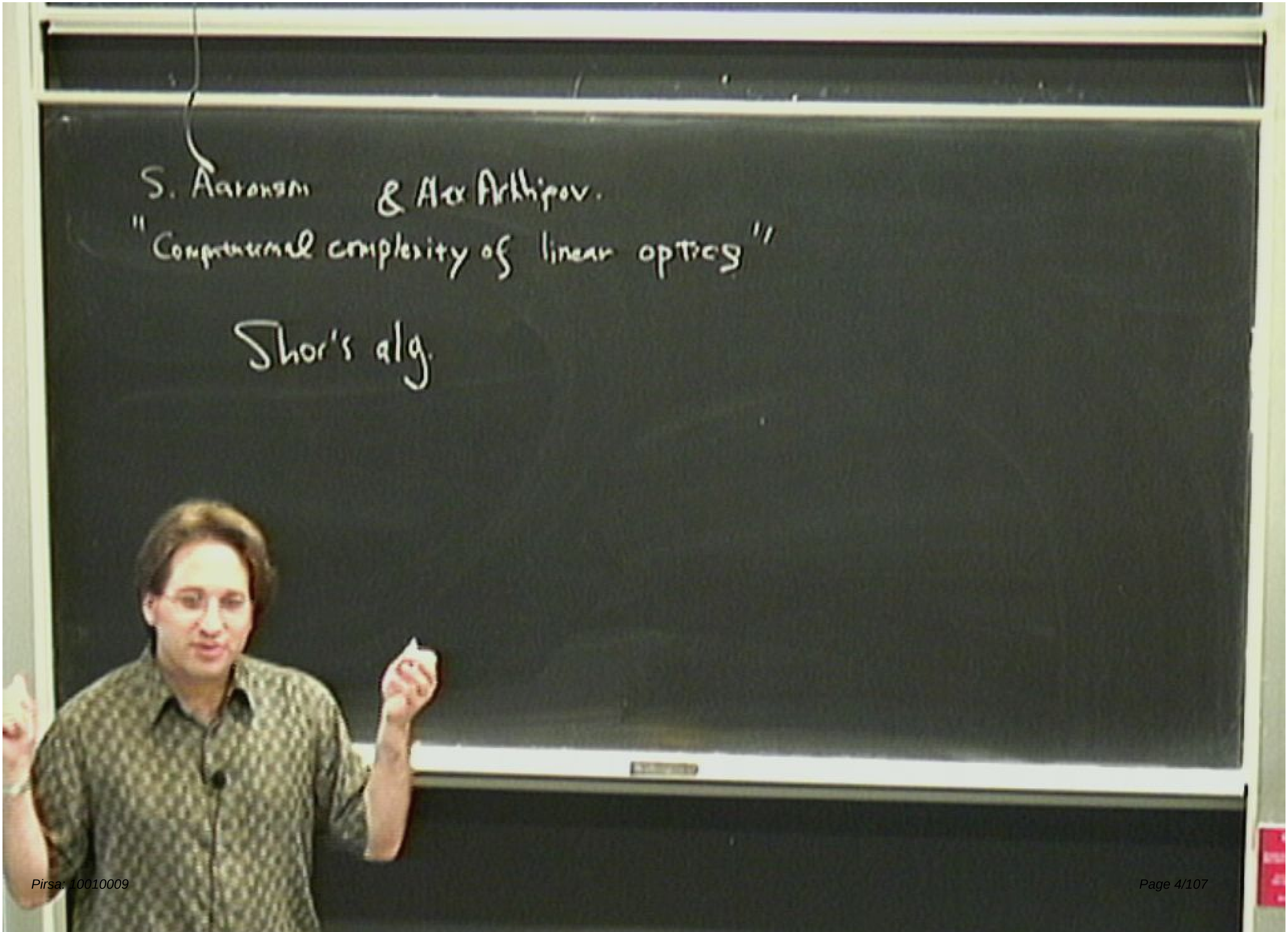
URL: http://pirsa.org/10010009

Abstract: I'll discuss some work-in-progress about the computational complexity of simulating the extremely "simple" quantum systems that arise in linear optics experiments. I'll show that *either* one can describe an experiment, vastly easier than building a universal quantum computer, that would test whether Nature is performing nontrivial quantum computation, or else one can give surprising new algorithms for approximating the permanent. Audience feedback as to which of these possibilities is the right one is sought. Joint work with Alex Arkhipov.


S. Aaronson
"Computational complexity of linear optics"

S. Aaronson & Alex Arkhipov.

"Computational complexity of linear optics"

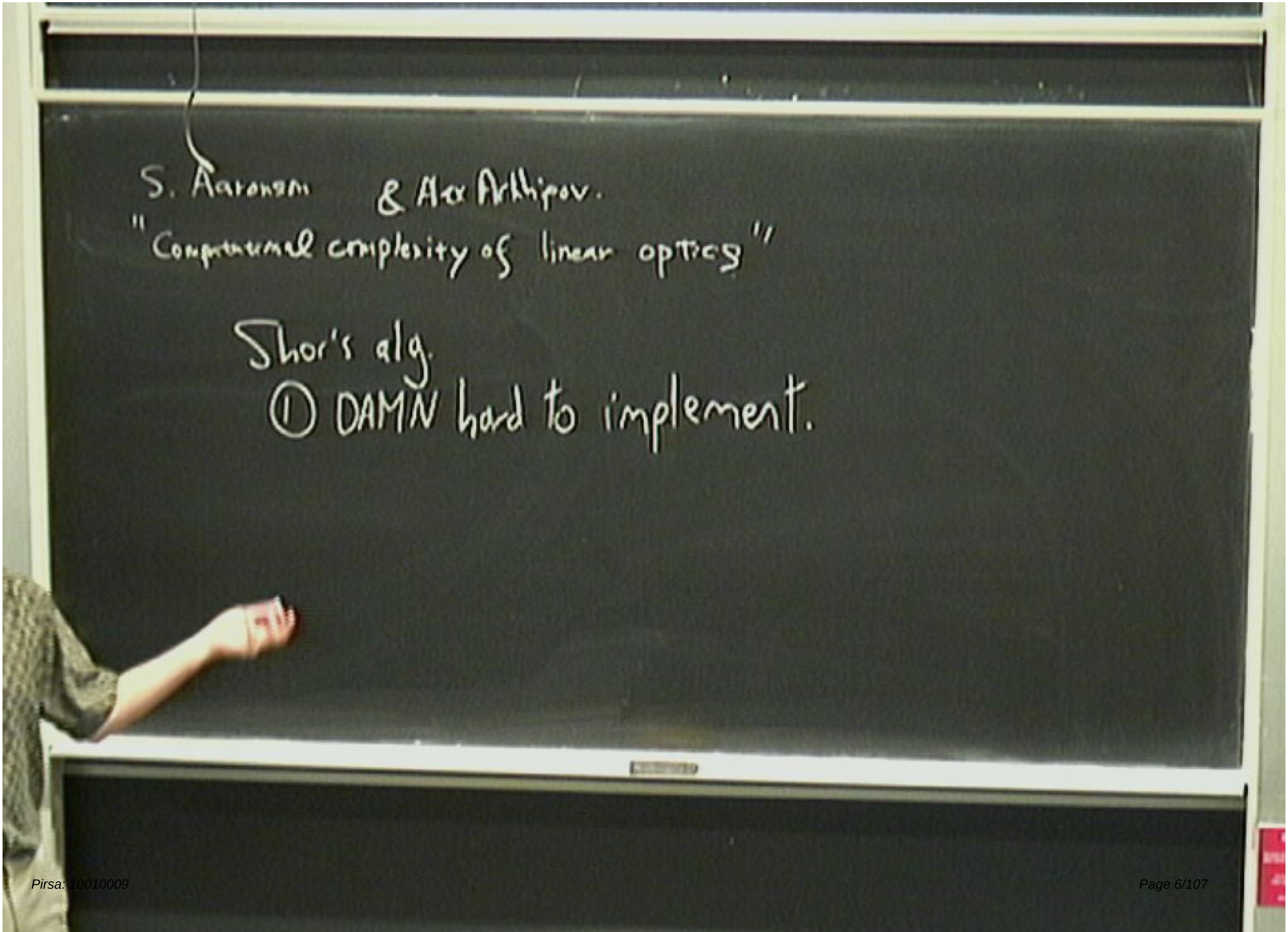

S. Aaronson & Alex Arkhipov.
"Computational complexity of linear optics"
Shor's alg.

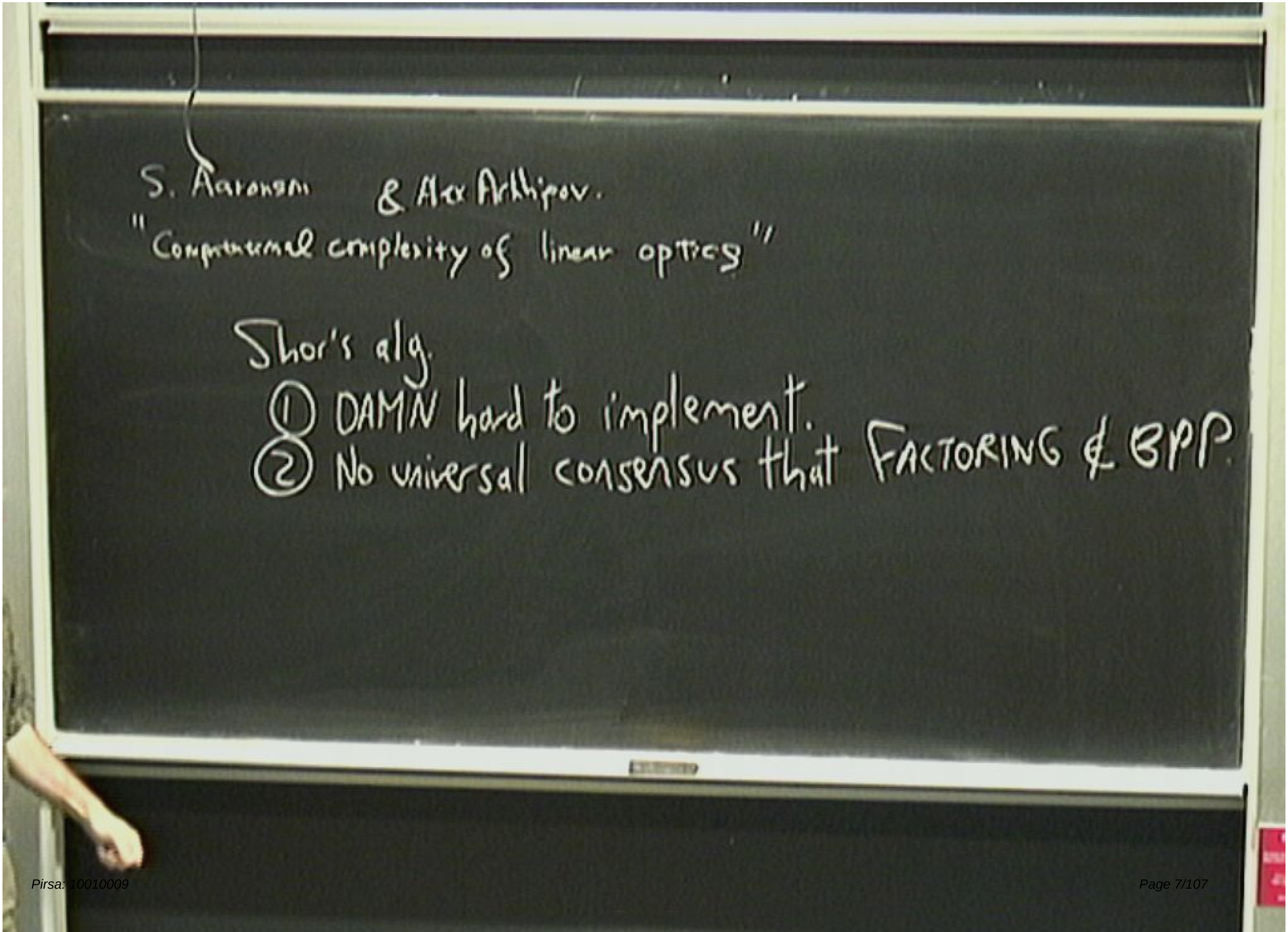
S. Aaronson & Alex Arkhipov.

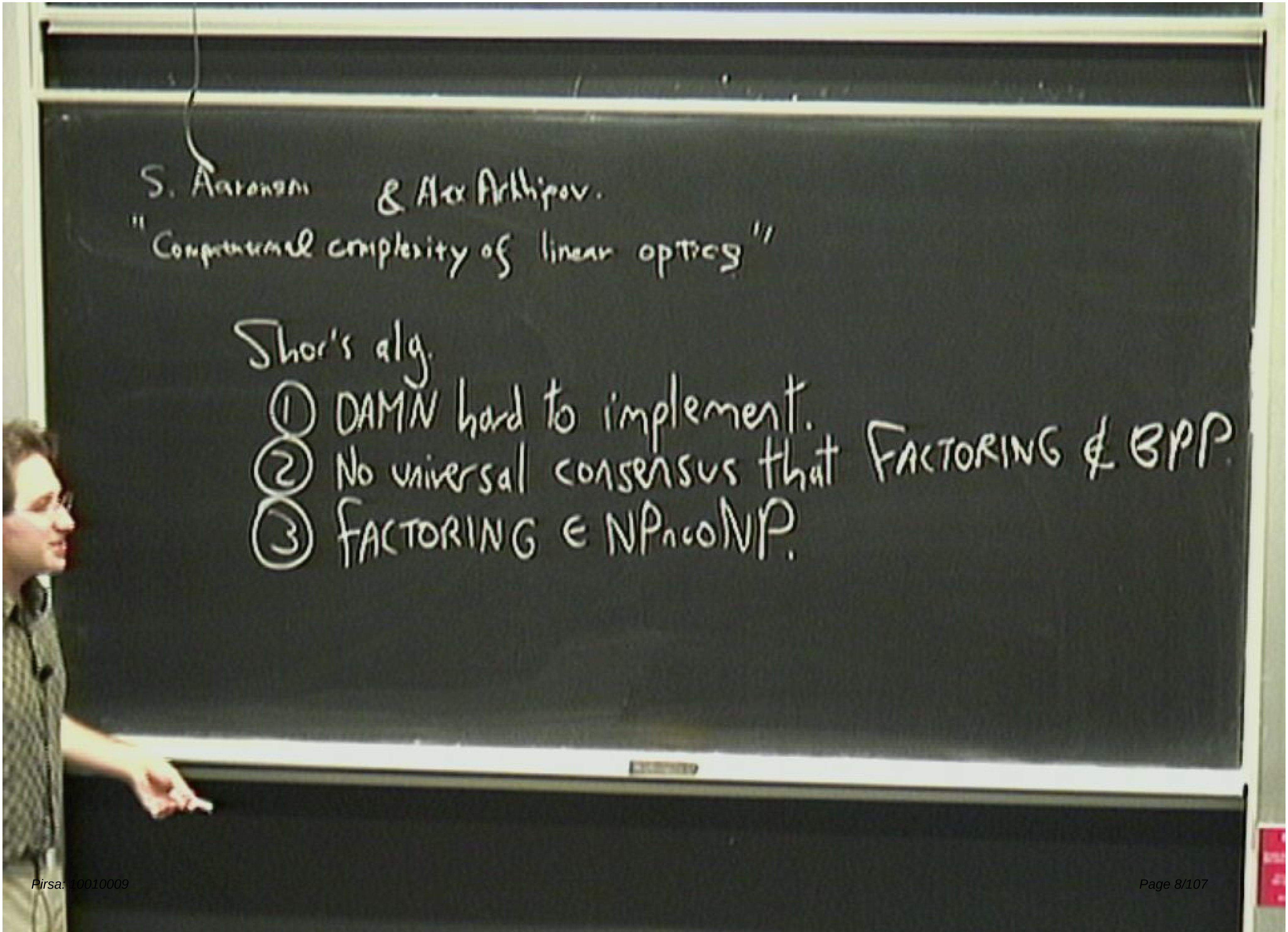
"Computational complexity of linear optics"

Shor's alg.

1. DAMN hard to implement.


S. Aaronson & Alex Arkhipov.
"Computational complexity of linear optics"
Shor's alg.
① DAMN hard to implement.


S. Aaronson & Alex Arkhipov.
"Computational complexity of linear optics"
Shor's alg.
① DAMN hard to implement.
② No universal consensus that FACTORING ∉ BPP.


S. Aaronson & Alex Arkhipov.
"Computational complexity of linear optics"
Shor's alg.
1 DAMN hard to implement.
2 No universal consensus that FACTORING ∉ BPP.
3 FACTORING ∈ NP∩coNP.

S. Aaronson & Alex Arkhipov.

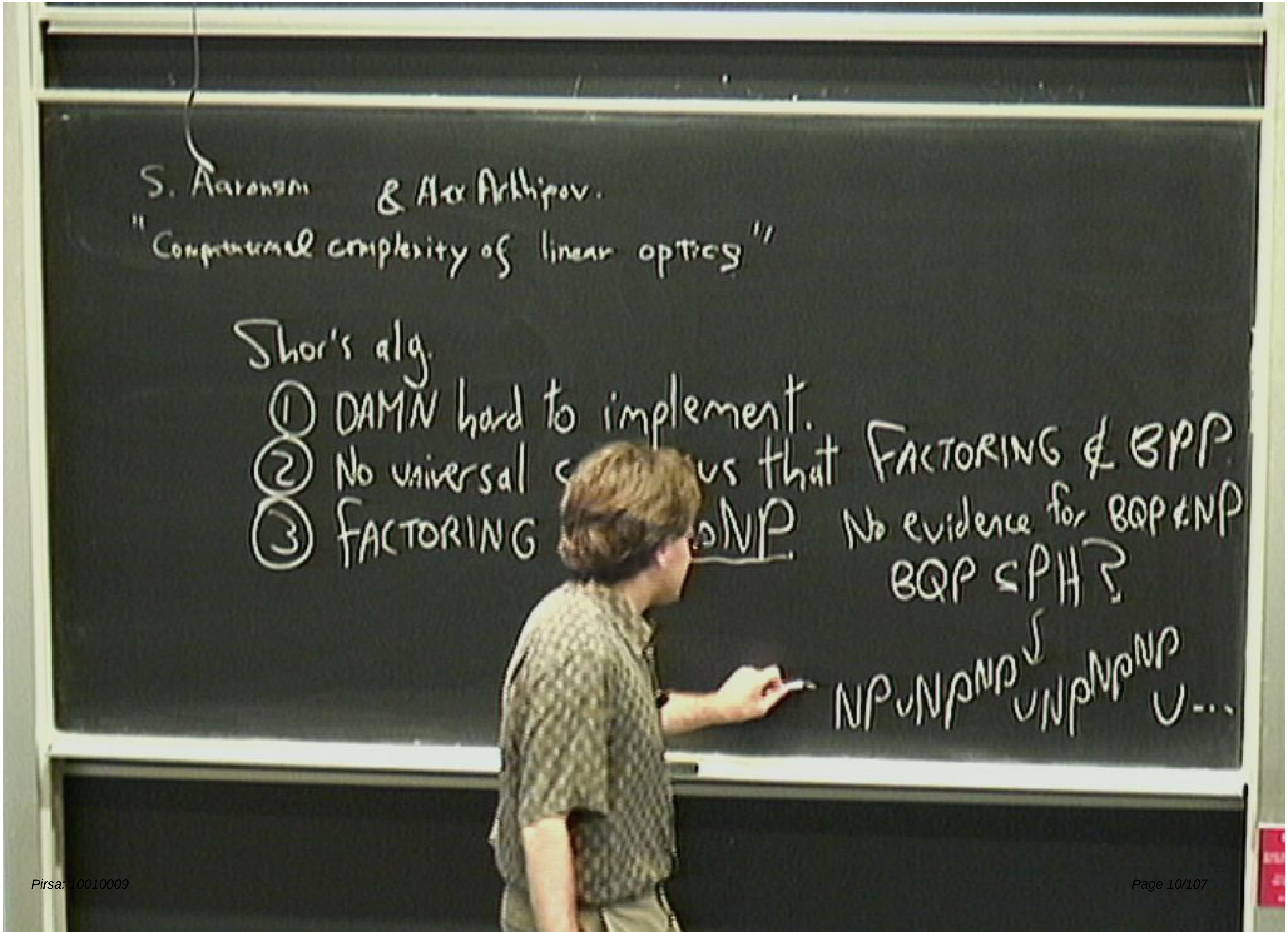
"Computational complexity of linear optics"

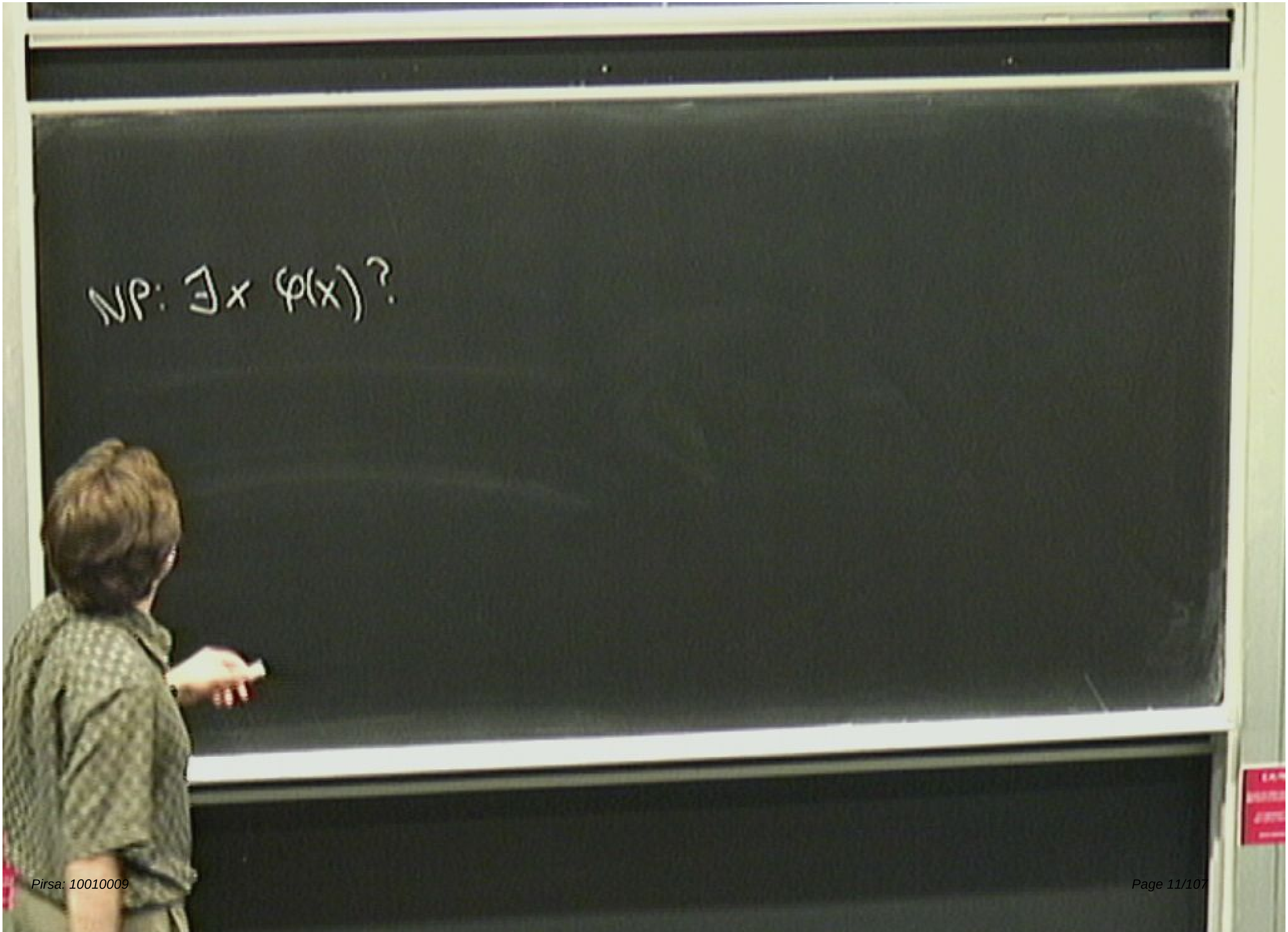
Shor's alg.

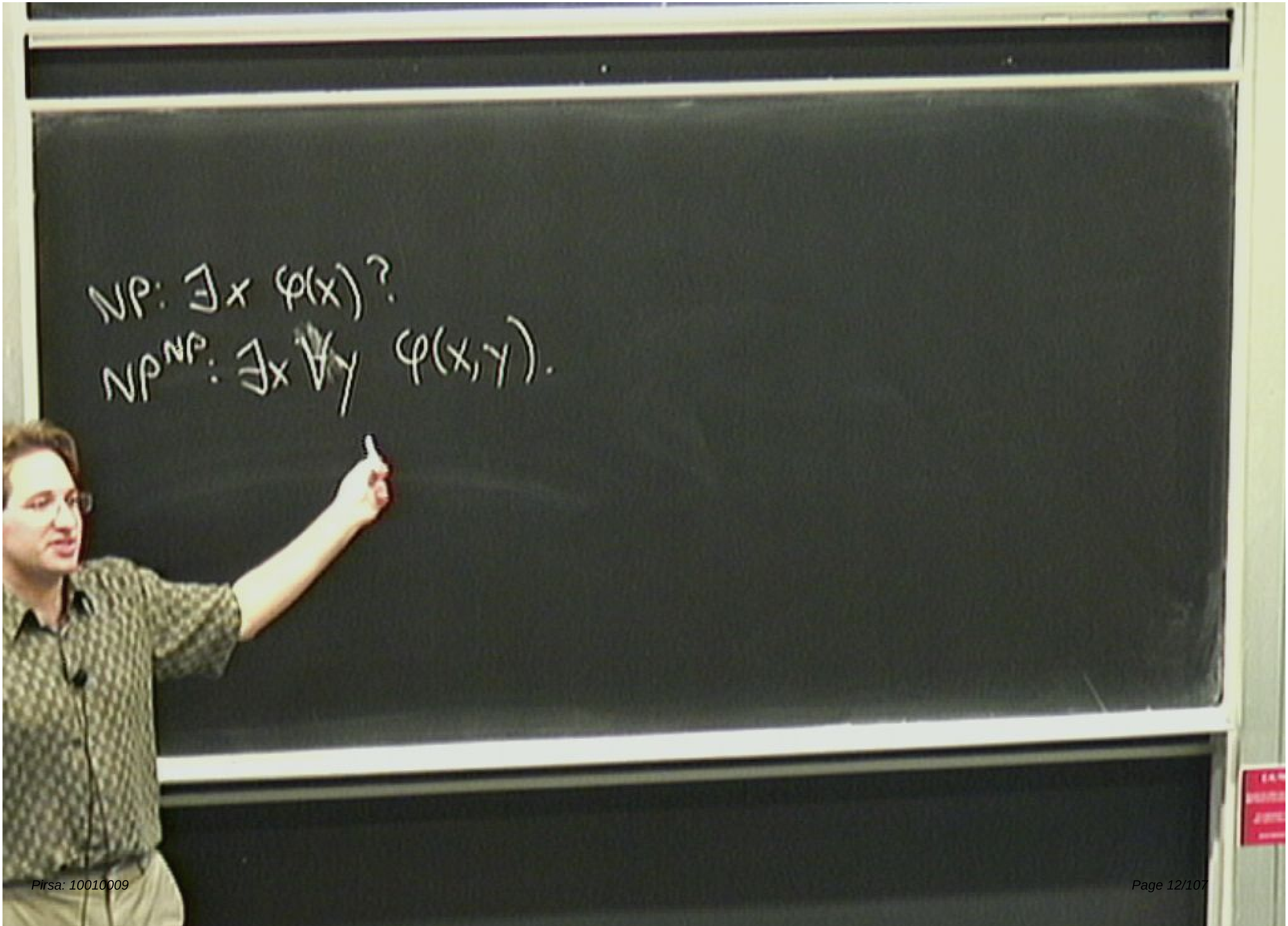
1. DAMN hard to implement.
2. No universal consensus that FACTORING ∉ BPP.
3. FACTORING ∈ $\underline{NP \cap coNP}$. No evid... $BQP \not\subset NP$

BQP... ?

N...


S. Aaronson & Alex Arkhipov.
"Computational complexity of linear optics"
Shor's alg.
① DAMN hard to implement.
② No universal s
us that FACTORING ∉ BPP
③ FACTORING
NP
No evidence for BQP ⊄ NP
BQP ⊆ PH ?
NP ∪ NP^NP ∪ NP^NP^NP ∪ ...


NP: $\exists x \ \varphi(x)$?


NP: $\exists x\ \varphi(x)$?
NP^{NP}: $\exists x \forall y\ \varphi(x,y)$.

$NP: \exists x\ \varphi(x)?$

$NP^{NP}: \exists x \forall y\ \varphi(x,y).$

$NP^{NP^{NP}}: \exists x \forall y \exists z\ \varphi(x,y,z)$

NP: $\exists x\ \varphi(x)$?

NP^{NP}: $\exists x\ \forall y\ \varphi(\underline{x, y})$.

$NP^{NP^{NP}}$: $\exists x\ \forall y\ \exists z\ \varphi(x, y, z)$

"Such-and-such is true $\Rightarrow$ PH collapses to a finite level"

NP: $\exists x\ \varphi(x)$?

NP^{NP}: $\exists x\ \forall y\ \varphi(\underline{x, y})$.

$NP^{NP^{NP}}$: $\exists x\ \forall y\ \exists z\ \varphi(x, y, z)$

"Such-and-such is true $\Rightarrow$ PH collapses to a finite level"

NP: $\exists x\ \varphi(x)$?

NP^{NP}: $\exists x\ \forall y\ \varphi(x,y)$.

$NP^{NP^{NP}}$: $\exists x\ \forall y\ \exists z\ \varphi(x,y,z)$

$\exists x$! $\exists$

"Such-and-such is true $\Rightarrow$ PH collapses to a finite level"

$P = NP \Rightarrow P = PH$.

$NP^{NP^{NP}} = NP^{NP}$

$$P = NP \Rightarrow P = PH.$$

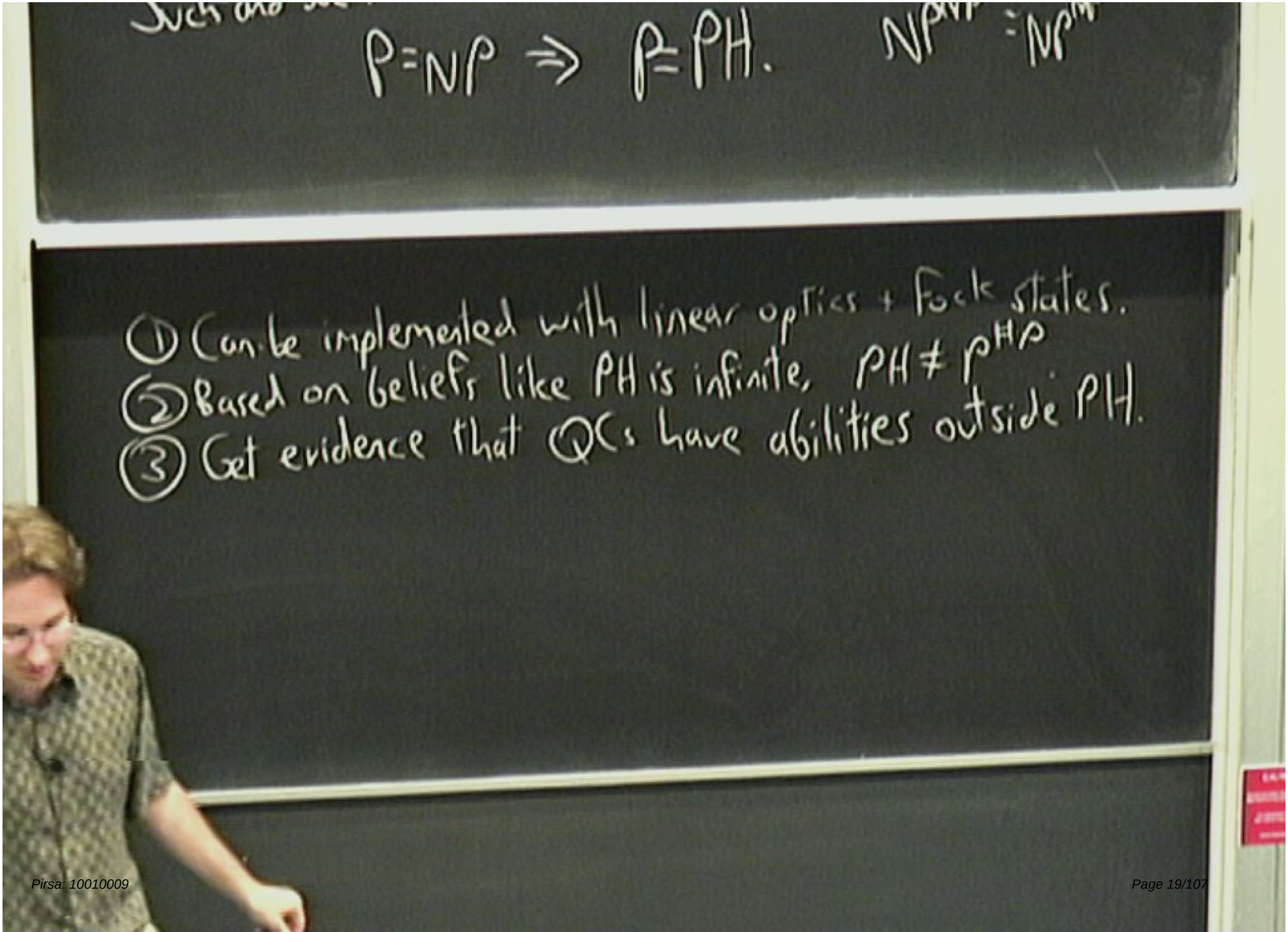
① Can be implemented with linear optics + Fock states.

Such and su...

$$P = NP \Rightarrow P = PH. \qquad NP^{NP} = NP^{NP}$$

① Can be implemented with linear optics + Fock states.

② Based on beliefs like PH is infinite, $PH \neq P^{\#P}$.


P=NP ⇒ P=PH.
① Can be implemented with linear optics + Fock states.
② Based on beliefs like PH is infinite, PH ≠ P^#P
③ Get evidence that QCs have abilities outside PH.

Such and su...

$$P = NP \Rightarrow P = PH. \qquad NP^{\dots} = NP^{\dots}$$

Adv:

① Can be implemented with linear optics + Fock states.

② Based on beliefs like PH is infinite, $PH \neq P^{\#P}$

③ Get evidence that QCs have abilities outside PH.

Disadv:

① Sampling & relational problems, not decision probs.

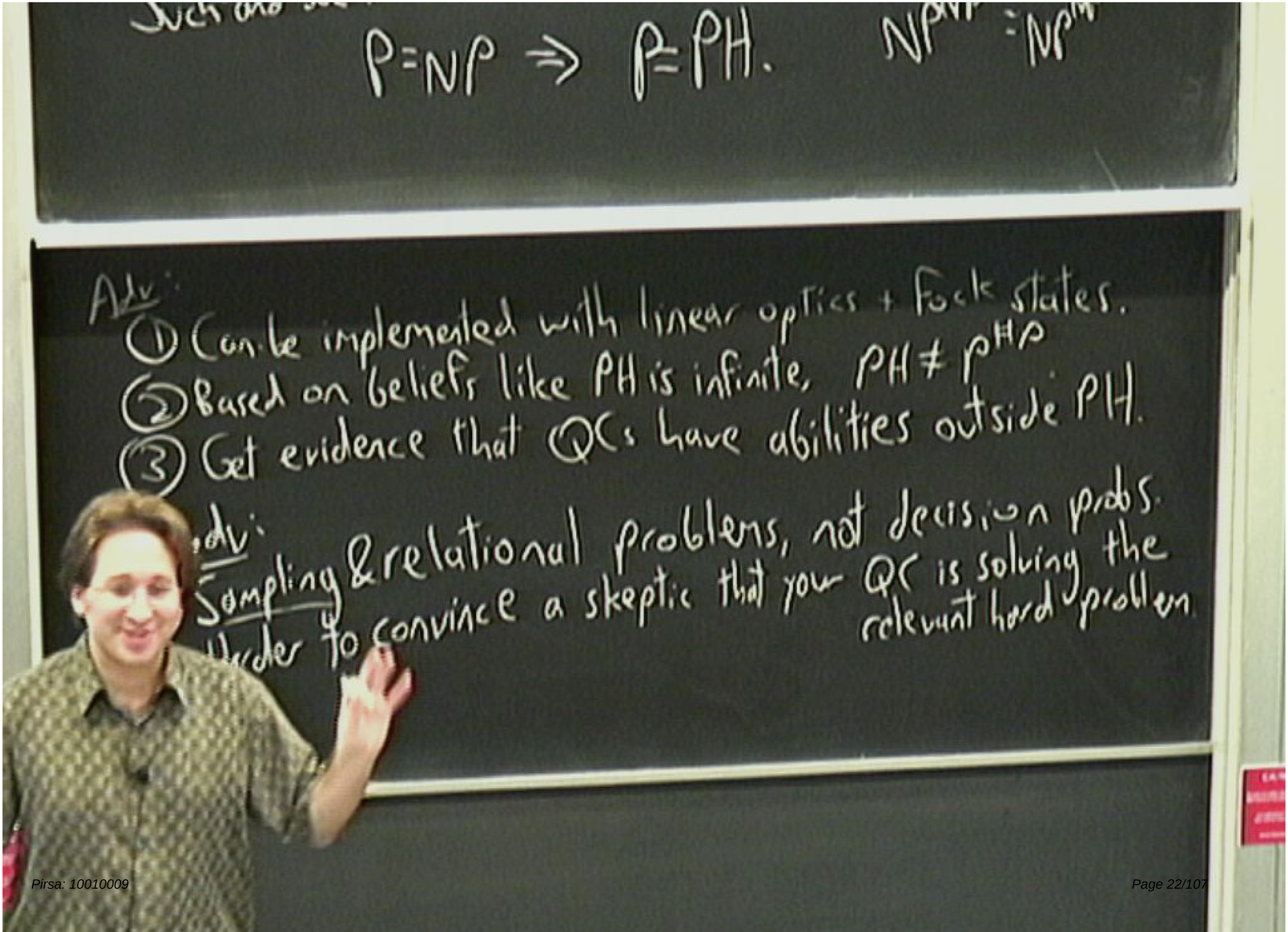
$$P = NP \Rightarrow P = PH. \qquad NP^{NP} = NP^{\dots}$$

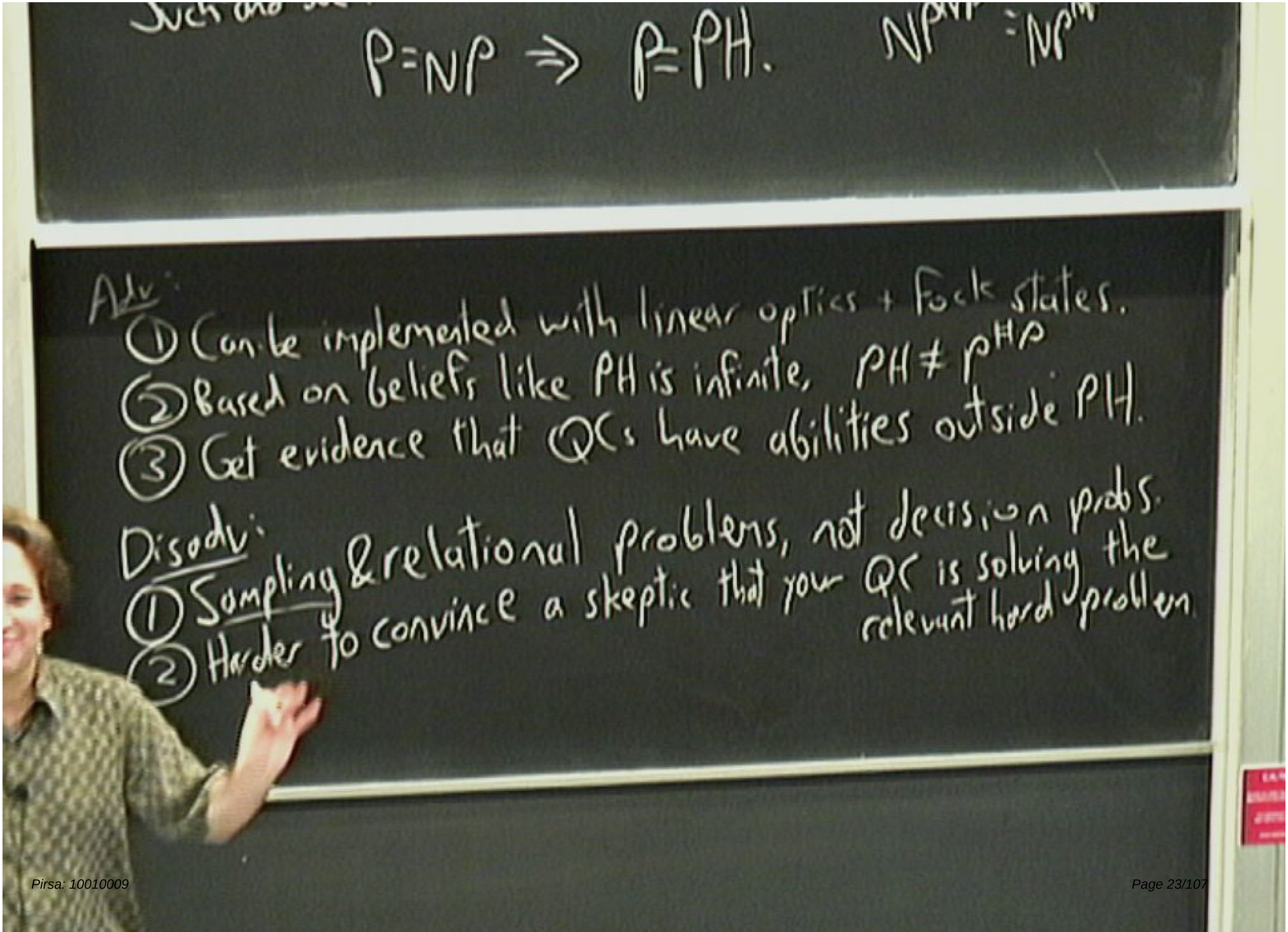
Adv:

1. Can be implemented with linear optics + Fock states.
2. Based on beliefs like PH is infinite, $PH \neq P^{\#P}$
3. Get evidence that QCs have abilities outside PH.

Disadv:

1. Sampling & relational problems, not decision probs.
2. Harder to convince a skeptic that your QC is solving the relevant hard problem.


P=NP ⇒ P=PH.
Adv:
① Can be implemented with linear optics + Fock states.
② Based on beliefs like PH is infinite, PH ≠ P#P
③ Get evidence that QCs have abilities outside PH.
Disadv:
Sampling & relational problems, not decision probs.
Harder to convince a skeptic that your QC is solving the relevant hard problem.


P=NP ⇒ P=PH.
Adv:
① Can be implemented with linear optics + Fock states.
② Based on beliefs like PH is infinite, PH ≠ P^#P
③ Get evidence that QCs have abilities outside PH.
Disadv:
① Sampling & relational problems, not decision probs.
② Harder to convince a skeptic that your QC is solving the relevant hard problem.

Such and su...

$$P = NP \Rightarrow P = PH. \qquad NP^{NP} = NP^{P}$$

Adv:

1. Can be implemented with linear optics + Fock states.
2. Based on beliefs like PH is infinite, $PH \neq P^{\#P}$
3. Get evidence that QCs have abilities outside PH.

Disadv:

1. Sampling & relational problems, not decision probs.
2. Harder to convince a skeptic that your QC is solving the relevant hard problem.

Such and

$$P = NP \Rightarrow P = PH. \qquad NP^{NP} = NP^{NP}$$

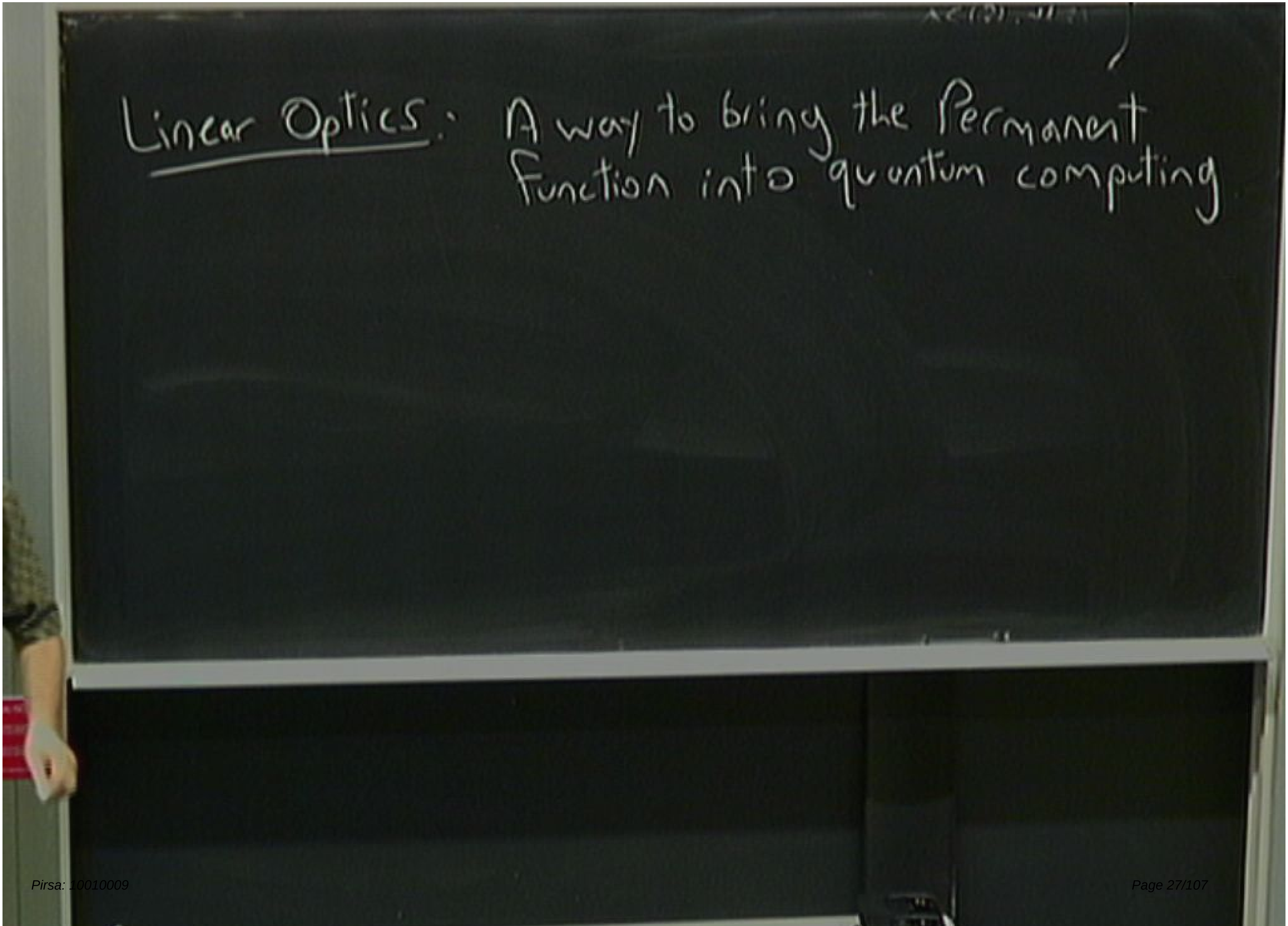
Adv:

1. Can be implemented with linear optics + Fock states.
2. Based on beliefs like PH is infinite, $PH \neq P^{\#P}$
3. Get evidence that QCs have abilities outside PH.

Disadv:

1. Sampling & relational problems, not decision probs.
2. Harder to convince a skeptic that your QC is solving the relevant hard problem.
3. Problems not "useful"?

Linear Optics.


Linear Optics: A way to bring the Permanent
function into quantum computing

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant $\quad Det(A) = \sum_{\sigma \in S_n} (-1)^{sgn(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$,

Bosons $\longleftrightarrow$ Permanent $\quad Per(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P) $\mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

$\searrow$ #P-complete.

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P) $\quad \mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\quad \mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

#P-complete.

Toda: $PH \subseteq P^{\#P}$

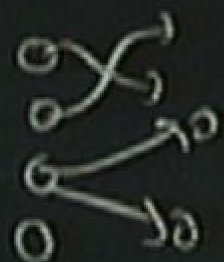
Valiant "Matchgate circuits"

Valiant "Matchgate circuits"

"Non-interacting fermions"

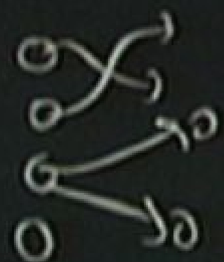
Valiant "Matchgate circuits"

"Non-interacting fermions"

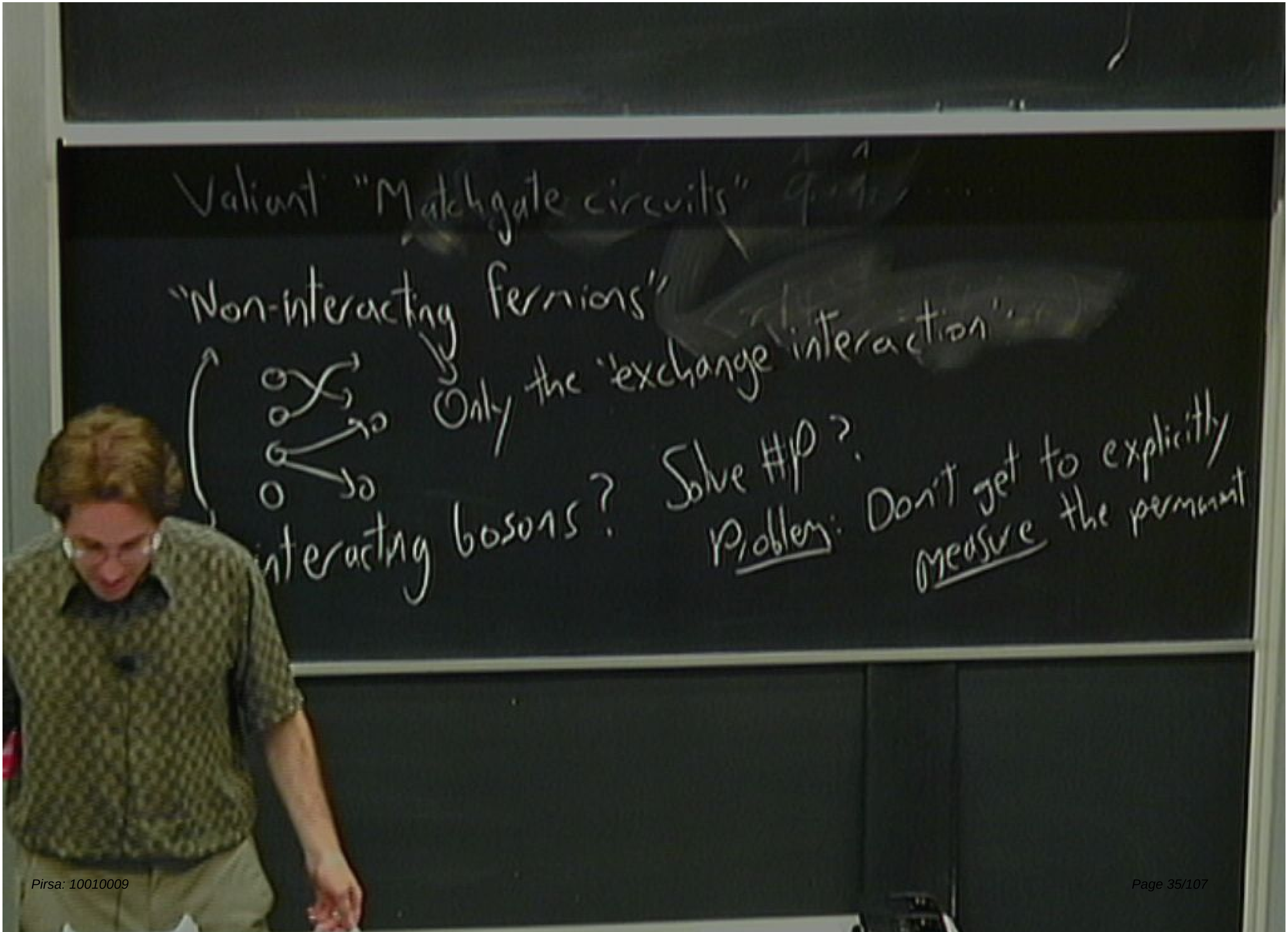


Valiant "Matchgate circuits"

"Non-interacting fermions"

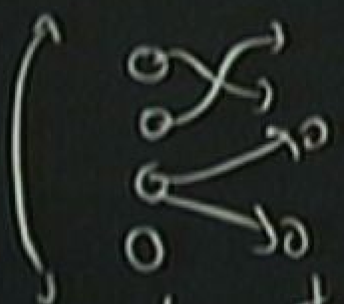


Only the "exchange interaction"


Valiant "Matchgate circuits"
"Non-interacting fermions"
Only the "exchange interaction"
interacting bosons? Solve #P?
Problem: Don't get to explicitly measure the permanent

Valiant "Matchgate circuits"

"Non-interacting fermions"



Only the "exchange interaction"

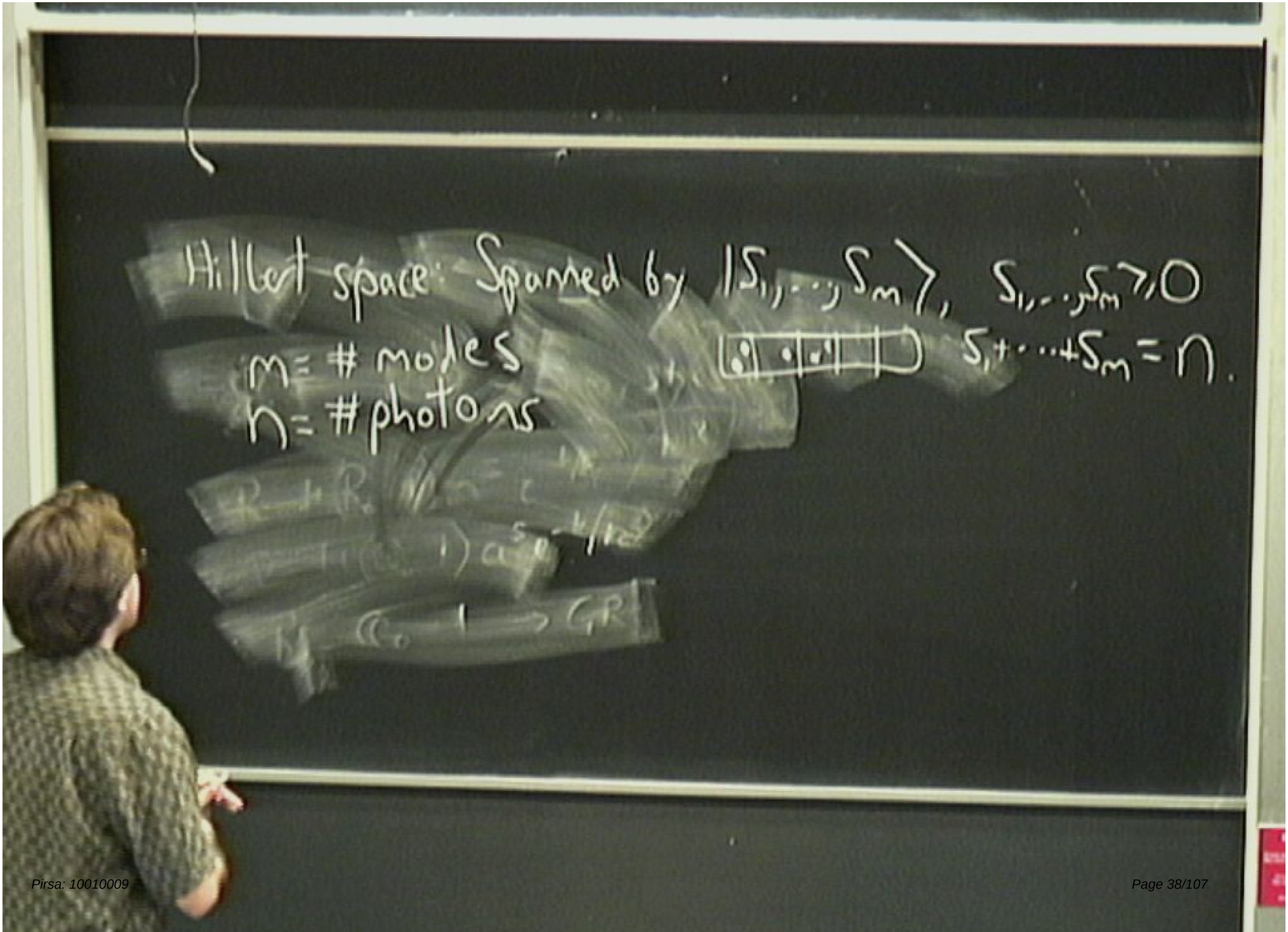
Non-interacting bosons? Solve #P?

Problem: Don't get to explicitly measure the permanent

Hilbert space: Spanned by $|s_1, \ldots, s_m\rangle$, $s_1, \ldots, s_m \geq 0$

$m = \#$ modes $\qquad \boxed{}$ $s_1 + \cdots + s_m = n$.

$n = \#$ photons


Hilbert space: Spanned by $|s_1, \ldots, s_m\rangle$, $s_1, \ldots, s_m \geq 0$
m = # modes
n = # photons
$s_1 + \cdots + s_m = n$.

Hilbert space: Spanned by $|S_1,\dots,S_m\rangle$, $S_1,\dots,S_m \geq 0$

m = # modes $\qquad$ $S_1+\dots+S_m = n$.

n = # photons $\qquad$ Initial state: $|1,\dots,1,0,\dots,0\rangle$ "$|I\rangle$".

Get to choose an $m \times m$ unitary U.

Hilbert space: Spanned by $|S_1,\ldots,S_m\rangle$, $S_1,\ldots,S_m \geq 0$

$\alpha_1 \; \alpha_2 \; \alpha_3 \; \alpha_4 \; \alpha_5$

m = # modes $\qquad S_1+\cdots+S_m = n$.

n = # photons $\qquad$ Initial state: $|1,\ldots,1,0,\ldots,0\rangle$ "$|I\rangle$".

Get to choose an $m \times m$ unitary U.

Hilbert space: Spanned by $|S_1,\dots,S_m\rangle$, $S_1,\dots,S_m \geq 0$

m = # modes $\qquad S_1+\dots+S_m = n$.

n = # photons $\qquad$ Initial state: $|1,\dots,1,0,\dots,0\rangle$ "$|I\rangle$".

Get to choose an $m \times m$ unitary U.

U induces an $\binom{m+n-1}{n} \times \binom{m+n-1}{n}$ unitary V as follows:

Hilbert space: Spanned by $|S_1,\ldots,S_m\rangle$, $S_1,\ldots,S_m \geq 0$

$S_1+\cdots+S_m = n$.

m = # modes

n = # photons

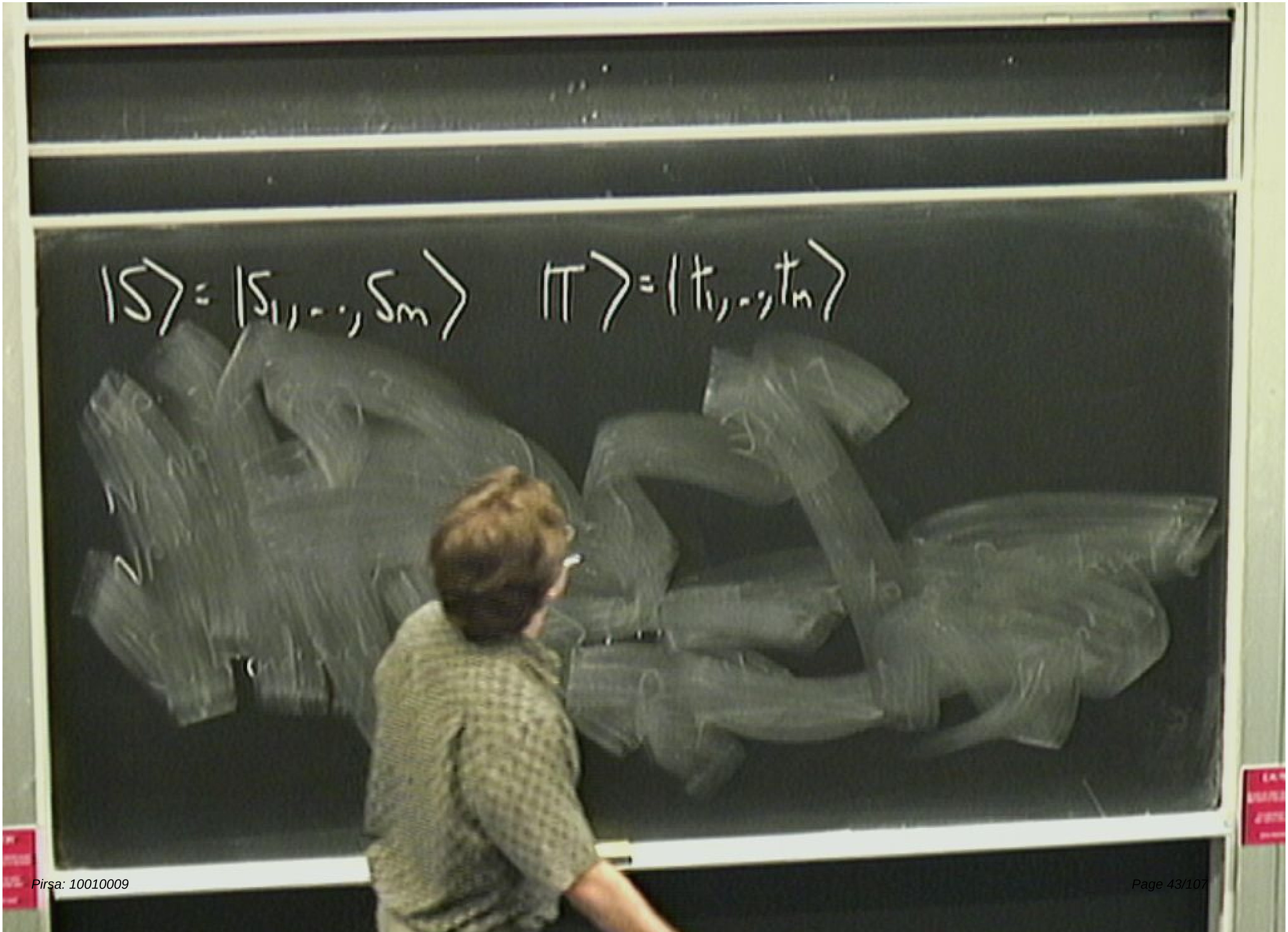
Initial state: $|1,\ldots,1,0,\ldots,0\rangle$ "$|I\rangle$".

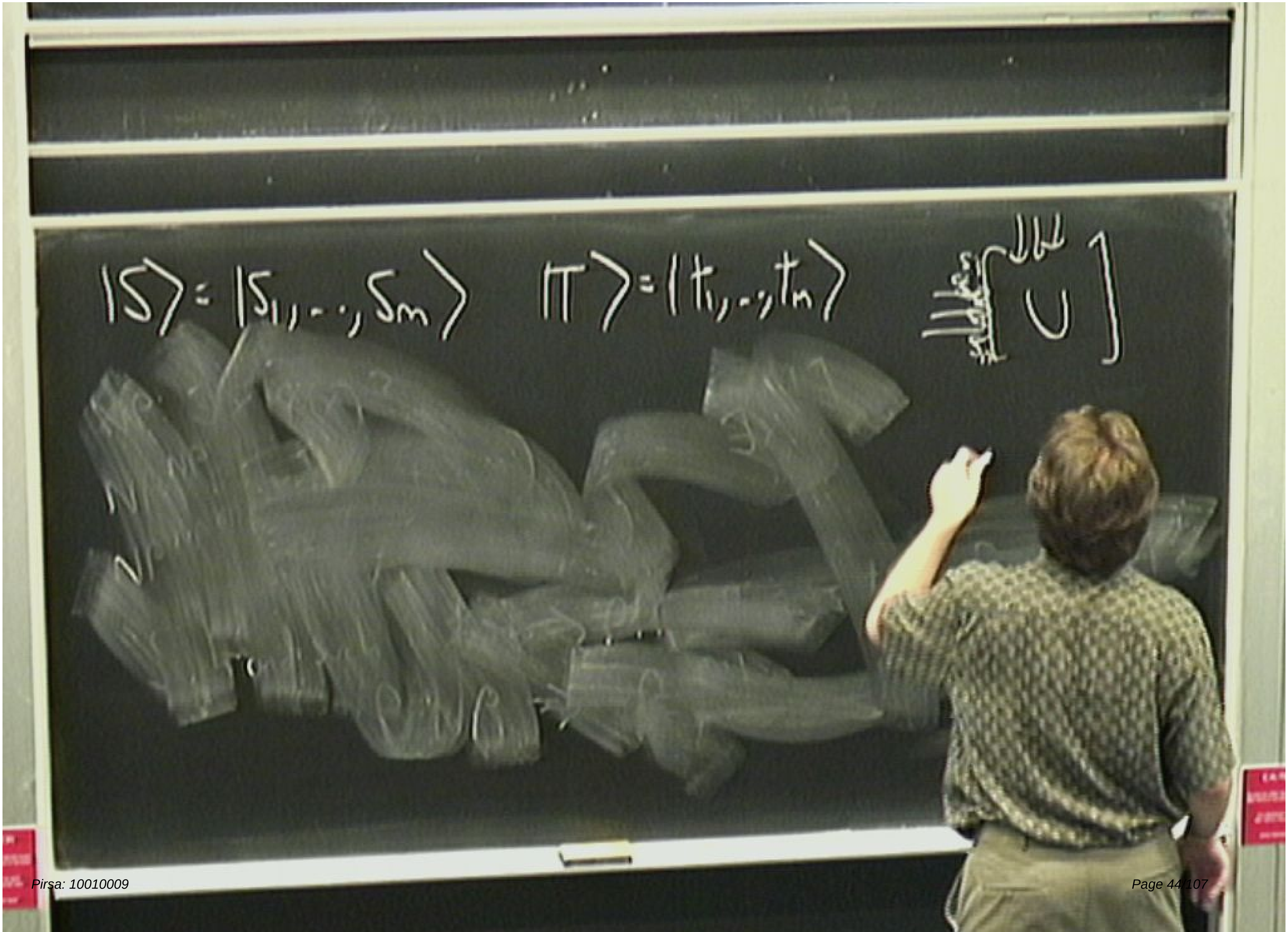
Get to choose an $m\times m$ unitary U.

U induces an $\binom{m+n-1}{n}\times\binom{m+n-1}{n}$ unitary V as follows:

$$\langle S|V|T\rangle = \frac{\mathrm{Per}(U_{S,T})}{\sqrt{s_1!\cdots s_m!\,t_1!\cdots t_m!}}$$

$U_{S,T}$ is an $n\times n$ submatrix of U.


$|S\rangle = |s_1, \ldots, s_m\rangle$
$|T\rangle = |t_1, \ldots, t_m\rangle$

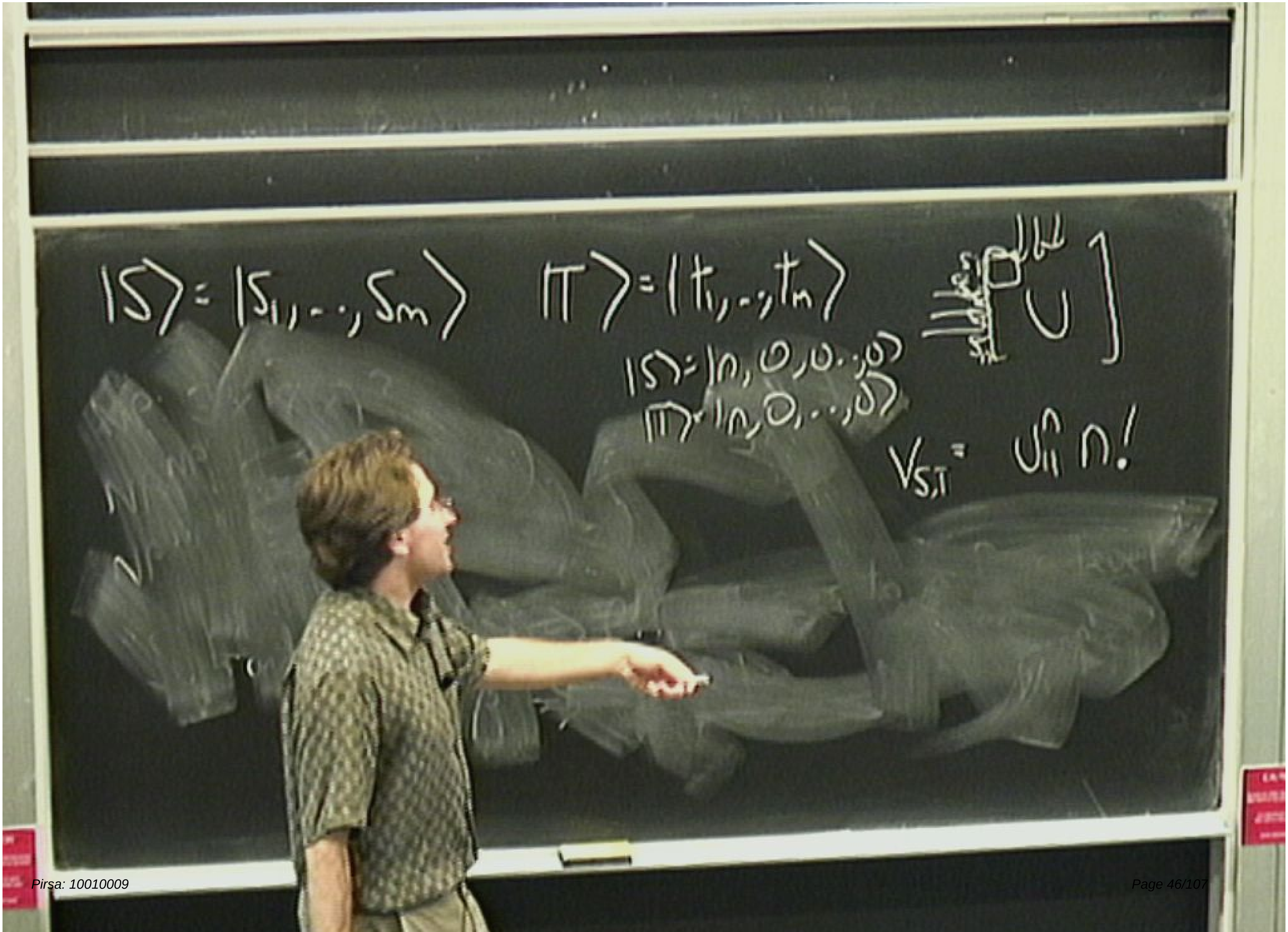

$|S\rangle = |s_1, \ldots, s_m\rangle$
$|T\rangle = |t_1, \ldots, t_n\rangle$
U

$|S\rangle = |s_1, \ldots, s_m\rangle \qquad |T\rangle = |t_1, \ldots, t_m\rangle$

$|n, 0, 0, \ldots, 0\rangle$

$|n, 0, \ldots, 0\rangle$

$[\,U\,]$


|S> = |s1, ..., sm>
|T> = |t1, ..., tm>
|S> = |n, 0, 0, ..., 0>
|T> = |n, 0, ..., 0>
VS,T = U11 n!

Hilbert space: Spanned by $|S_1, \ldots, S_m\rangle$, $S_1, \ldots, S_m \geq 0$

m = # modes $\quad S_1 + \cdots + S_m = n$.

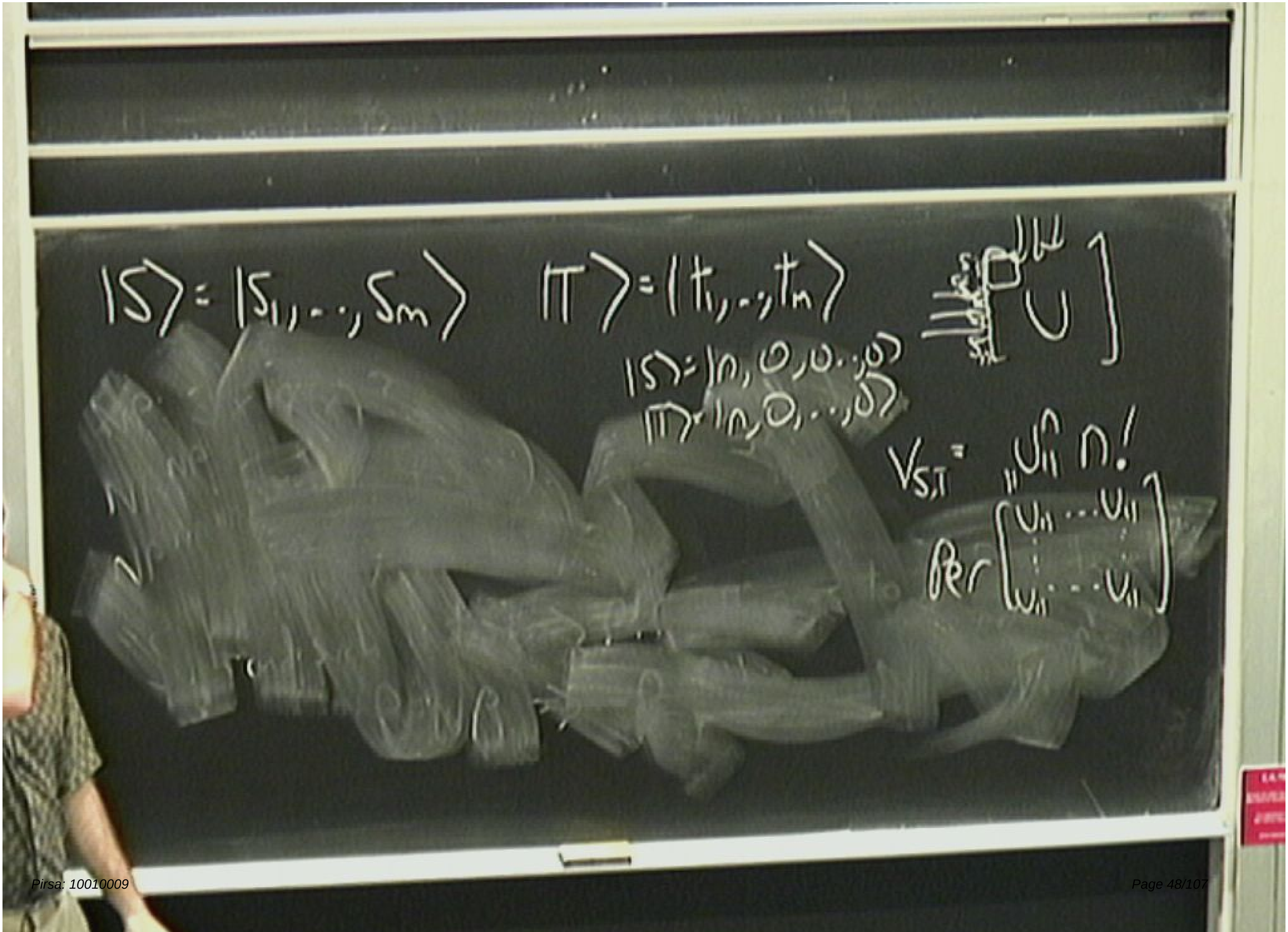
n = # photons $\quad$ Initial state: $|1, \ldots, 1, 0, \ldots, 0\rangle$ "$|I\rangle$".

Get to choose an $m \times m$ unitary U.

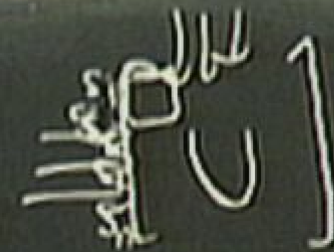
U induces an $\binom{m+n-1}{n} \times \binom{m+n-1}{n}$ unitary V as follows:

$$\langle S|V|T\rangle = \frac{\mathrm{Per}(U_{S,T})}{\sqrt{S_1! \cdots S_m!\, t_1! \cdots t_m!}}$$

$U_{S,T}$ is an $n \times n$ submatrix of U.


|S⟩ = |s1, ..., sm⟩
|T⟩ = |t1, ..., tm⟩
|S⟩ = |n, 0, 0, ..., 0⟩
|T⟩ = |n, 0, ..., 0⟩
V_S,T = u11^n n!
Per[u11 ... u11 / u11 ... u11]
U

$|S\rangle = |s_1, \dots, s_m\rangle \quad |T\rangle = |t_1, \dots, t_n\rangle$



Then, measure $V|I\rangle$
in computational basis
& output result

$|S\rangle = |n, 0, 0, \dots, 0\rangle$
$|T\rangle = |n, 0, \dots, 0\rangle$

$$V_{S,T} = u_{11}^n \, n!$$

$$\mathrm{Per}\begin{bmatrix} u_{11} & \cdots & u_{11} \\ \vdots & & \vdots \\ u_{11} & \cdots & u_{11} \end{bmatrix}$$

Hilbert space: Spanned by $|S_1, \ldots, S_m\rangle$, $S_1, \ldots, S_m \geq 0$

m = # modes $\qquad$ $S_1 + \cdots + S_m = n$.

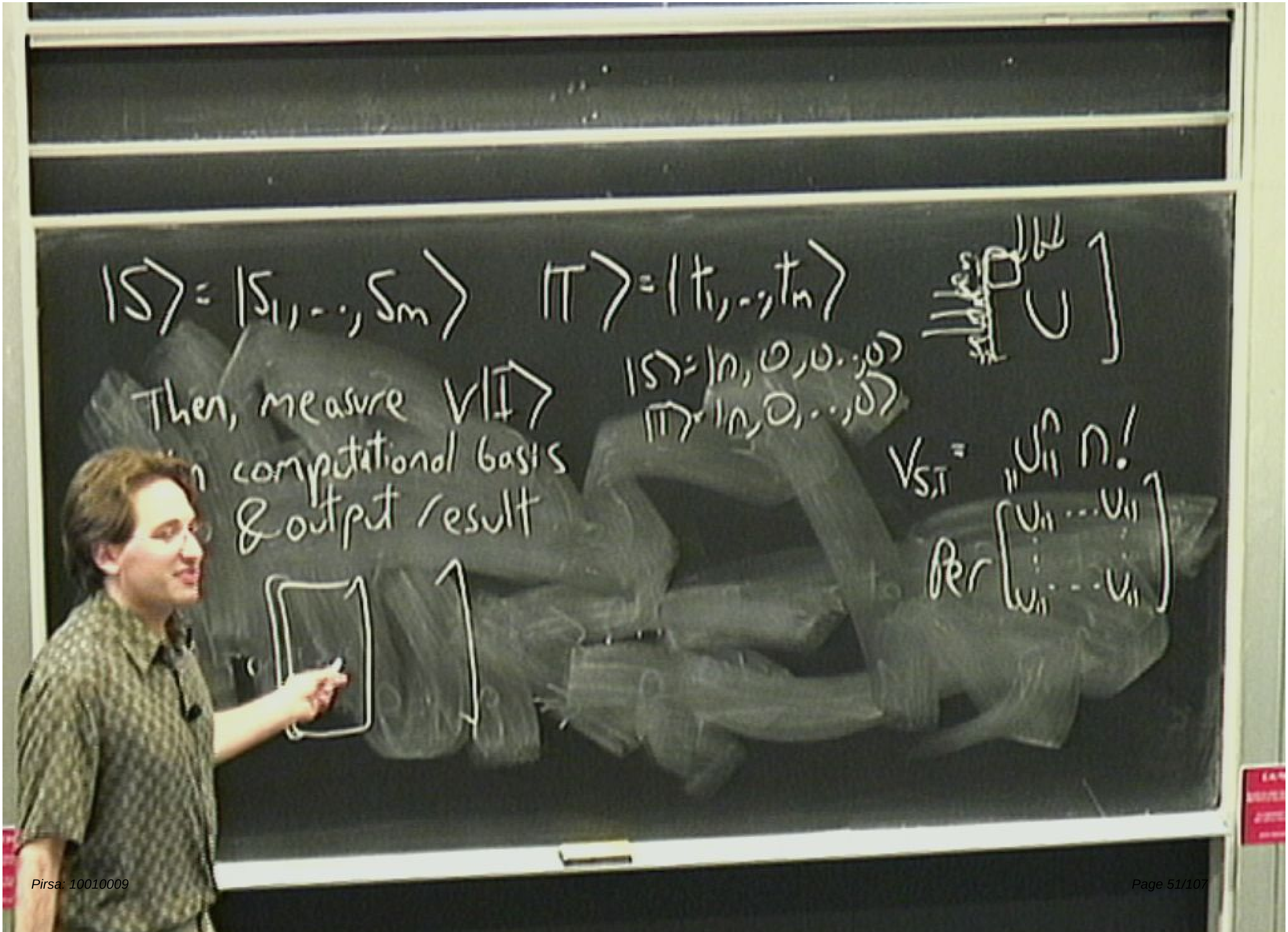
n = # photons $\qquad$ Initial state: $|1, \ldots, 1, 0, \ldots, 0\rangle$ "$|I\rangle$".

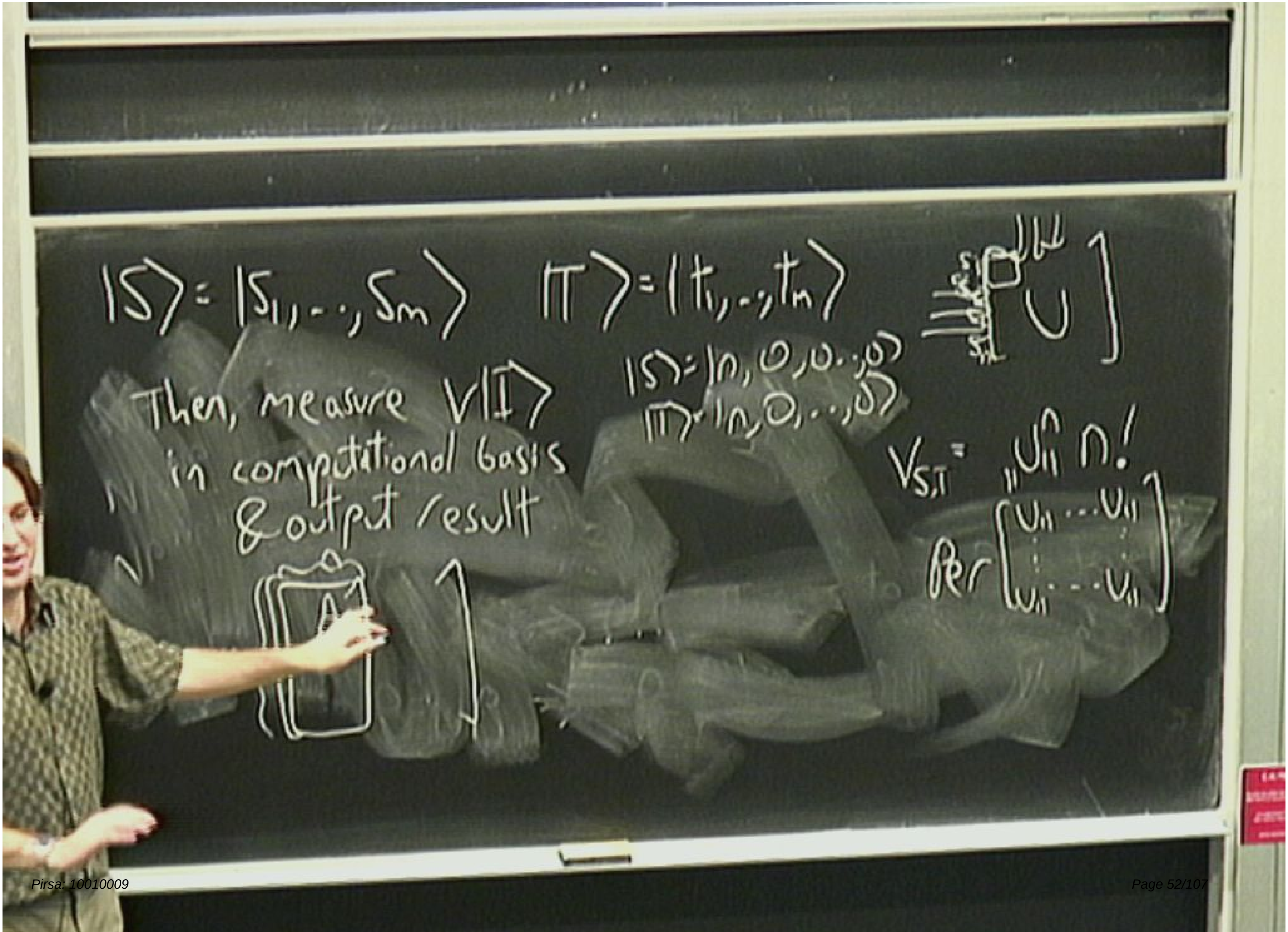
Get to choose an $m \times m$ unitary U.

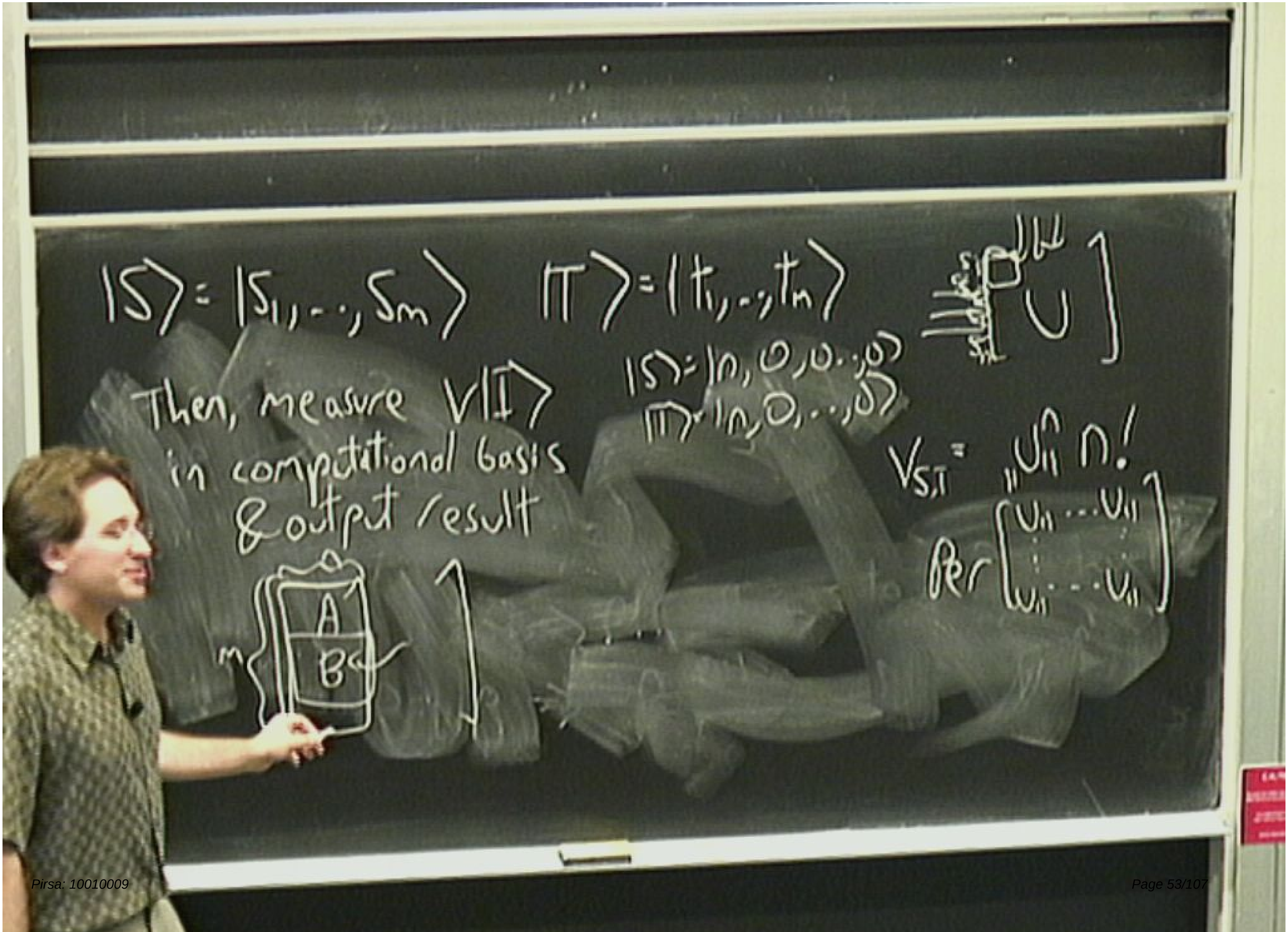
U induces an $\binom{m+n-1}{n} \times \binom{m+n-1}{n}$ unitary V as follows:

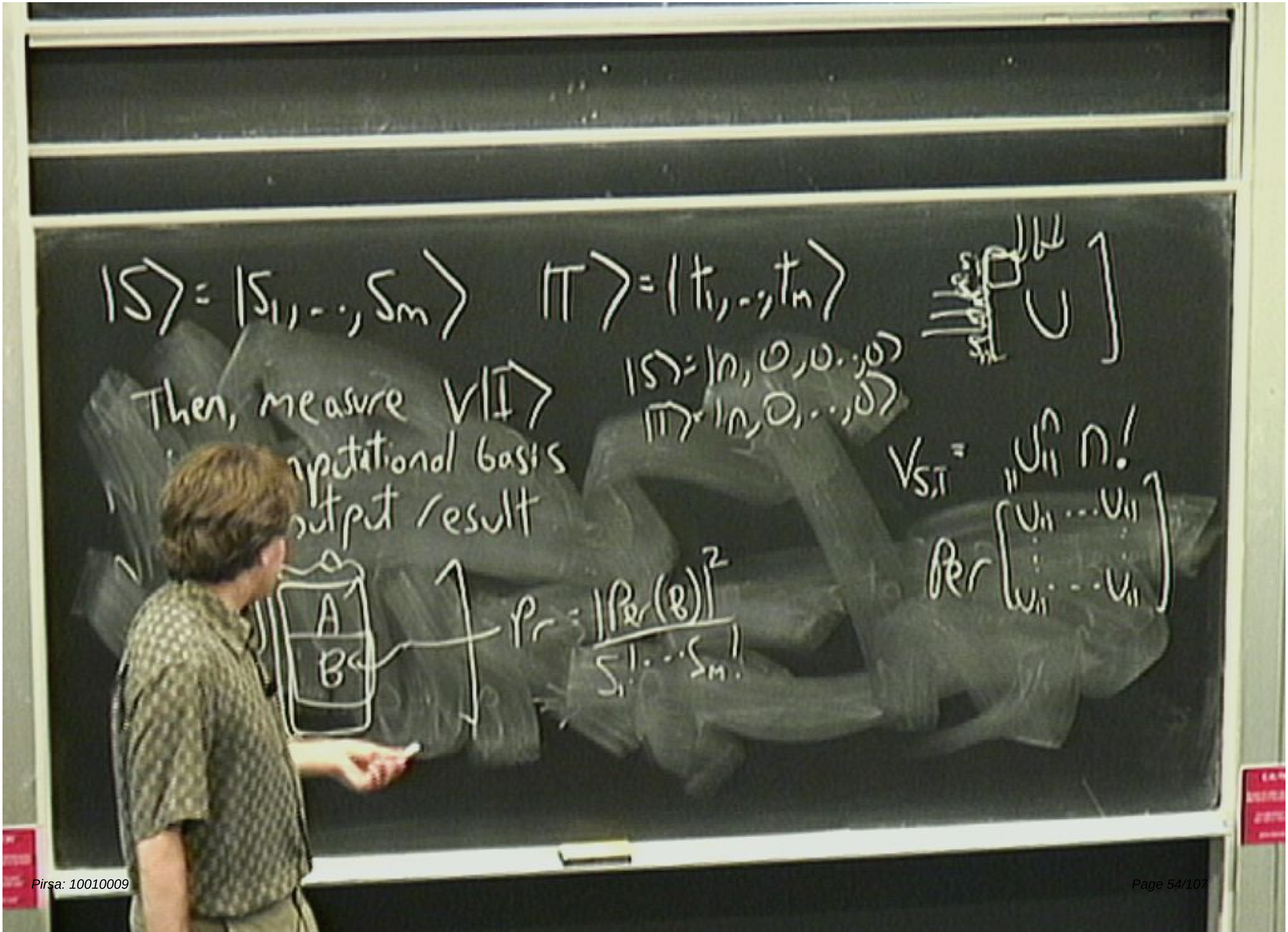
$$\langle S|V|T\rangle = \frac{\mathrm{Per}(U_{S,T})}{\sqrt{S_1! \cdots S_m! \, t_1! \cdots t_m!}}$$

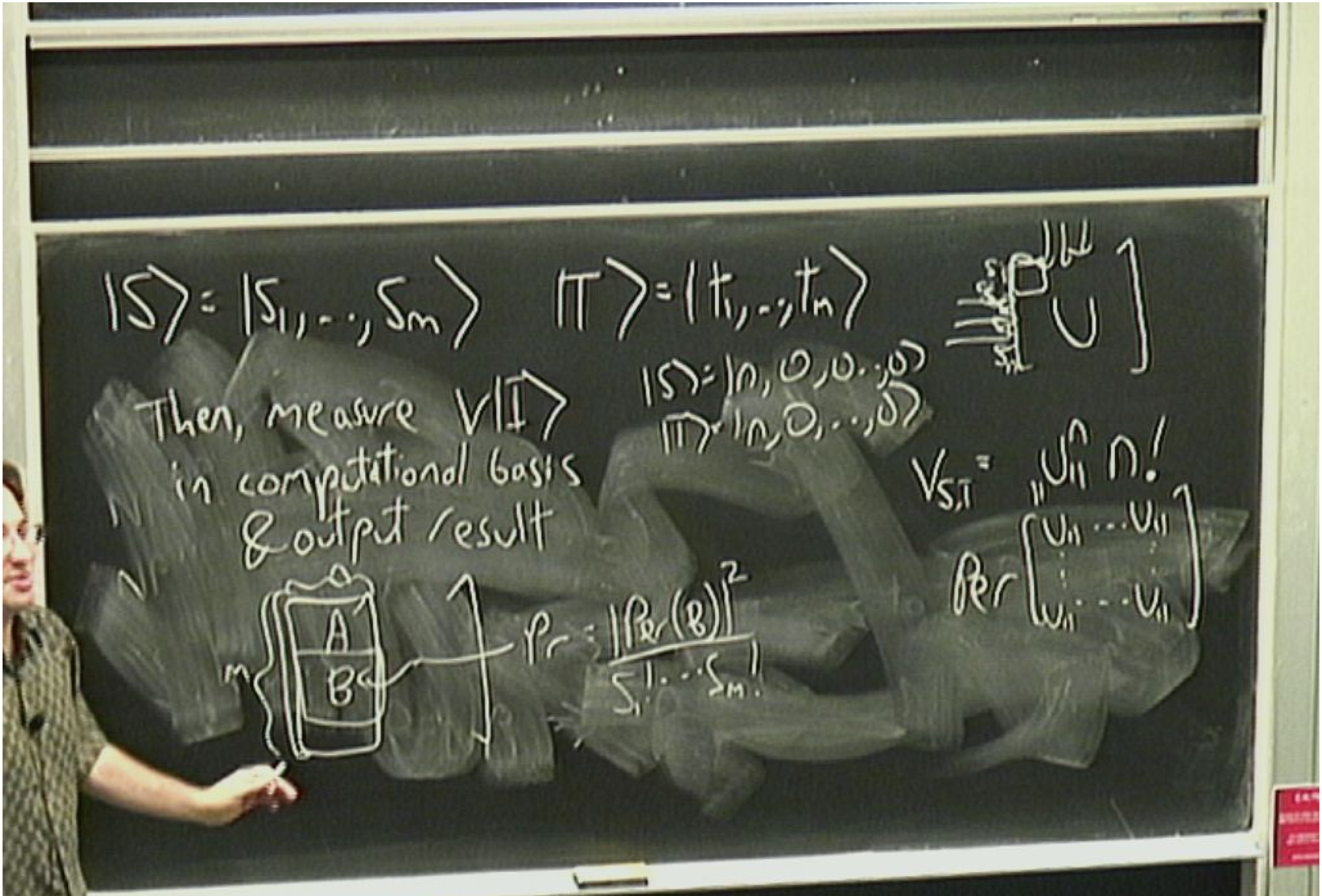
$U_{S,T}$ is an $n \times n$ submatrix of U.


|S⟩ = |s1, .., sm⟩
|T⟩ = |t1, .., tm⟩
Then, measure V|I⟩
in computational basis
& output result
|S⟩ = |n, 0, 0, .., 0⟩
|T⟩ = |n, 0, .., 0⟩
V_S,T = U11^n n!
Per


$|S\rangle = |s_1, \ldots, s_m\rangle$
$|T\rangle = |t_1, \ldots, t_m\rangle$
Then, measure $V|I\rangle$
in computational basis
& output result
$|S\rangle = |n, 0, 0, \ldots, 0\rangle$
$|T\rangle = |n, 0, \ldots, 0\rangle$
$V_{S,T} = u_{11}^n \, n!$
Per


|S> = |s1, ..., sm>
|T> = |t1, ..., tn>
Then, measure V|I>
in computational basis
& output result
A
B
m
U
|S> = |n, 0, 0, ..., 0>
|T> = |n, 0, ..., 0>
VS,T = u11^n n!
Per


|S⟩ = |s₁, ..., sₘ⟩
|T⟩ = |t₁, ..., tₙ⟩
U
Then, measure V|I⟩
computational basis
output result
|S⟩ = |n, 0, 0, .., 0⟩
|T⟩ = |n, 0, .., 0⟩
V_S,T = U₁₁ⁿ n!
Per
A
B
Pr = |Per(B)|² / s₁! ... sₘ!


|S> = |s1, ..., sm>
|T> = |t1, ..., tn>
U
Then, measure V|I>
in computational basis
& output result
|S> = |n, 0, 0, ..., 0>
|T> = |n, 0, ..., 0>
VS,T = U11^n n!
Per
A
B
m
Pr = |Per(B)|^2 / s1! ... sm!

$|S\rangle = |s_1, \ldots, s_m\rangle$ $\quad |T\rangle = |t_1, \ldots, t_m\rangle$

$|S\rangle = |n, 0, 0, \ldots, 0\rangle$

$|T\rangle = |n, 0, \ldots, 0\rangle$

Then, measure $V|I\rangle$ in computational basis & output result

$$m\left\{\begin{bmatrix} A \\ B \end{bmatrix}\right. \quad \Pr = \frac{|\mathrm{Per}(B)|^2}{s_1! \cdots s_m!}$$

$$V_{S,T} = U_{11}^n\, n! = \mathrm{Per}\begin{bmatrix} U_{11} & \cdots & U_{11} \\ \vdots & & \vdots \\ U_{11} & \cdots & U_{11} \end{bmatrix}$$

For us

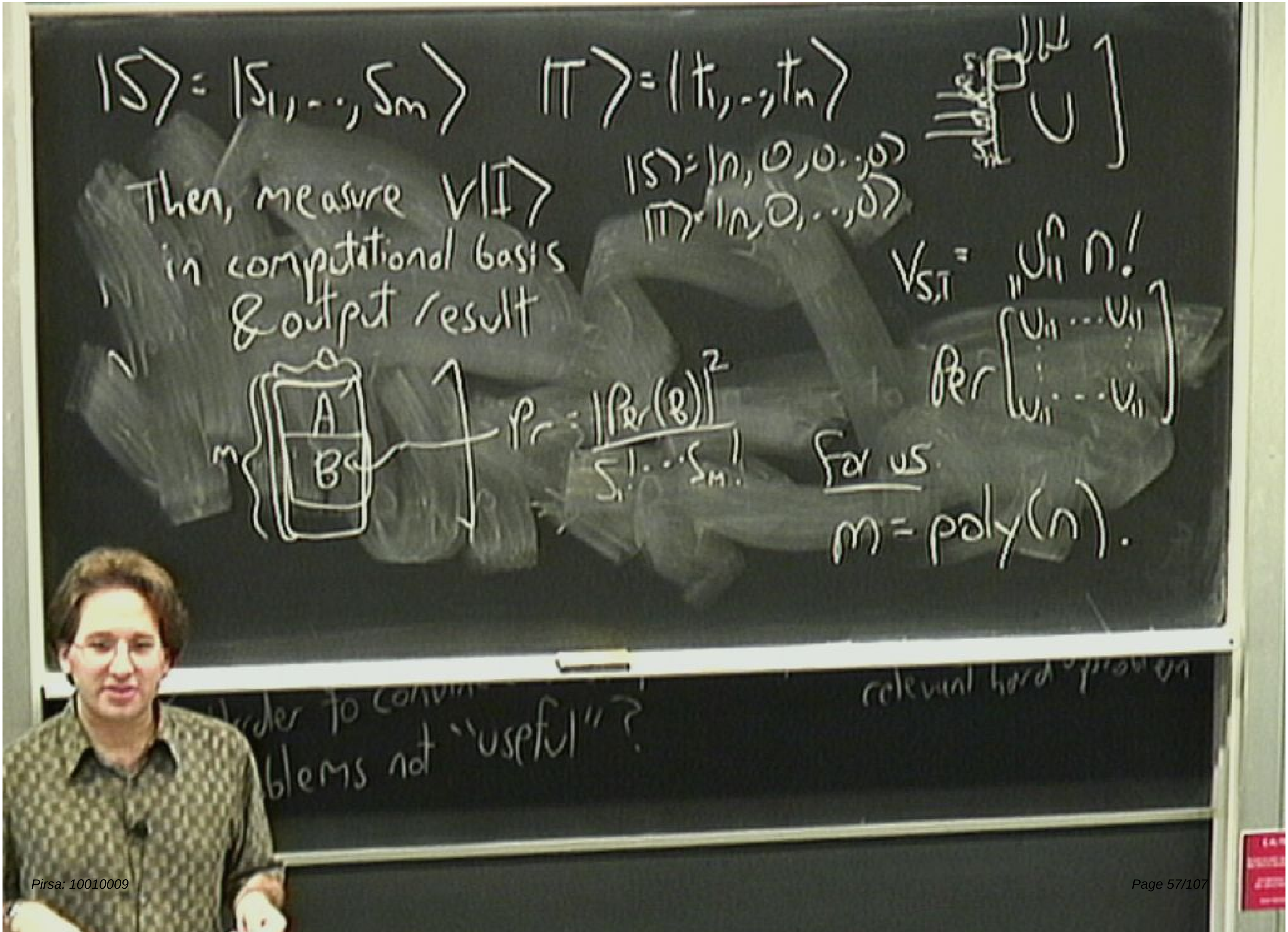
$m = \mathrm{poly}(n)$.

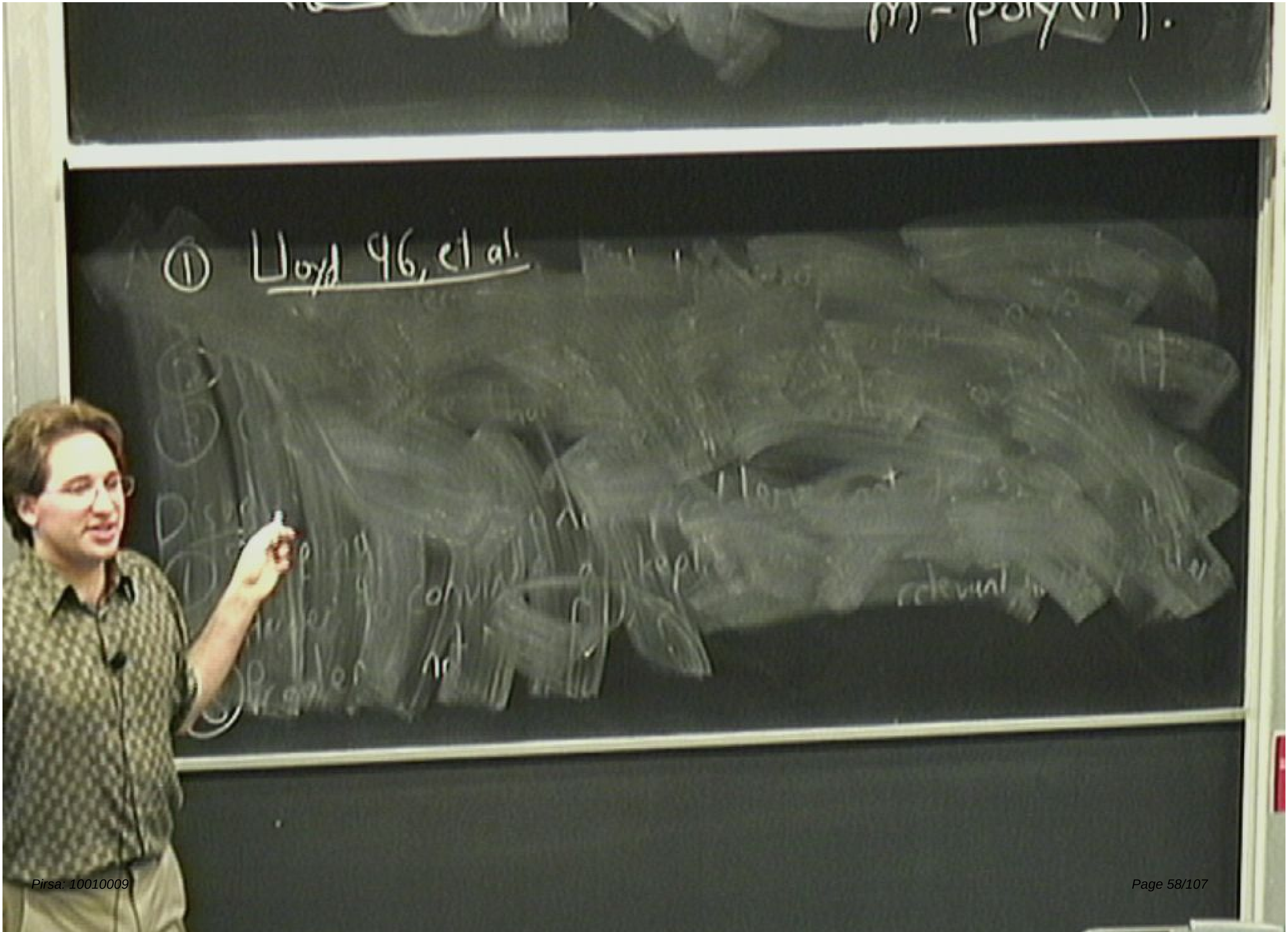
relevant hard problem

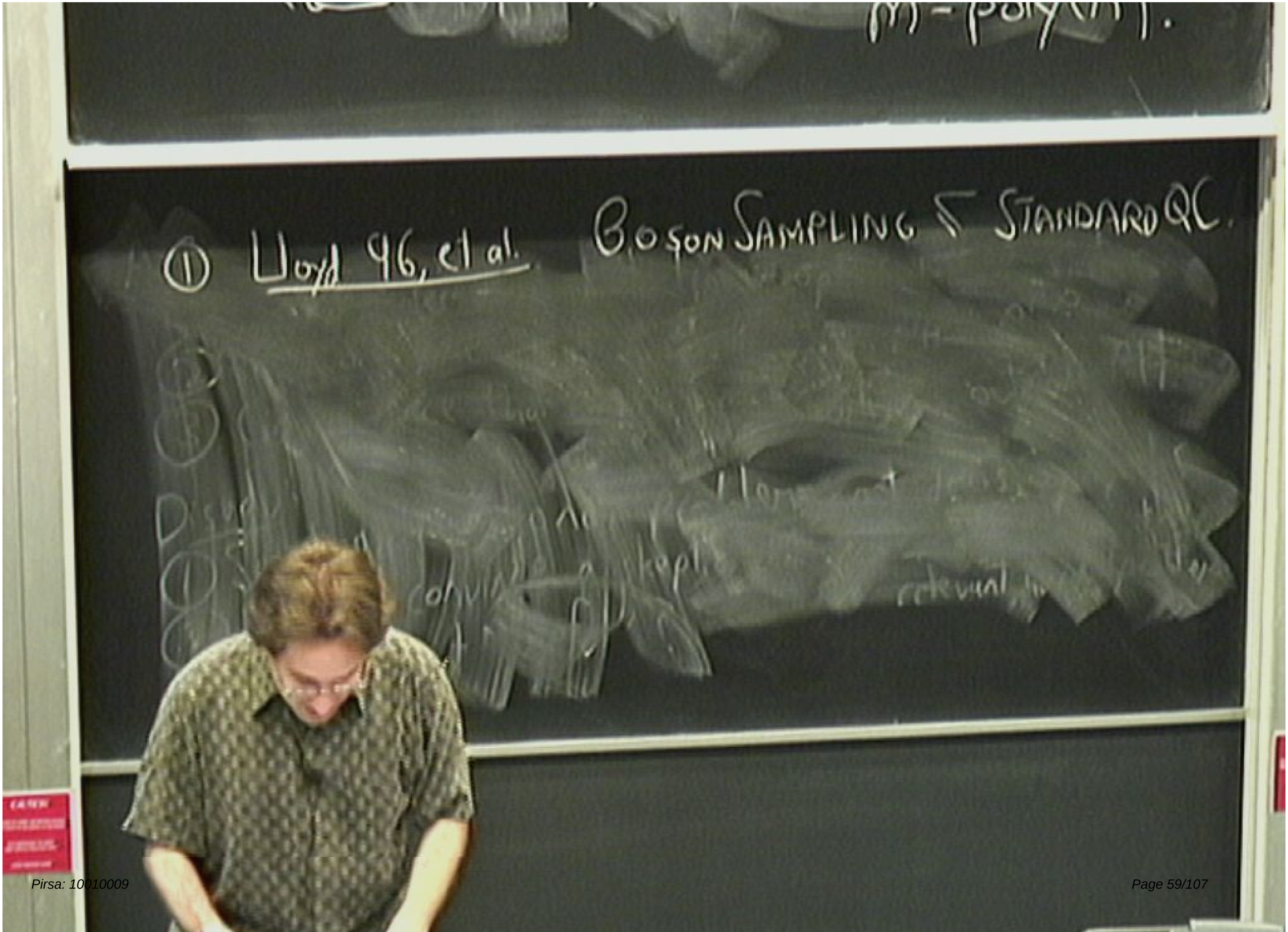
(2) Harder to convin...

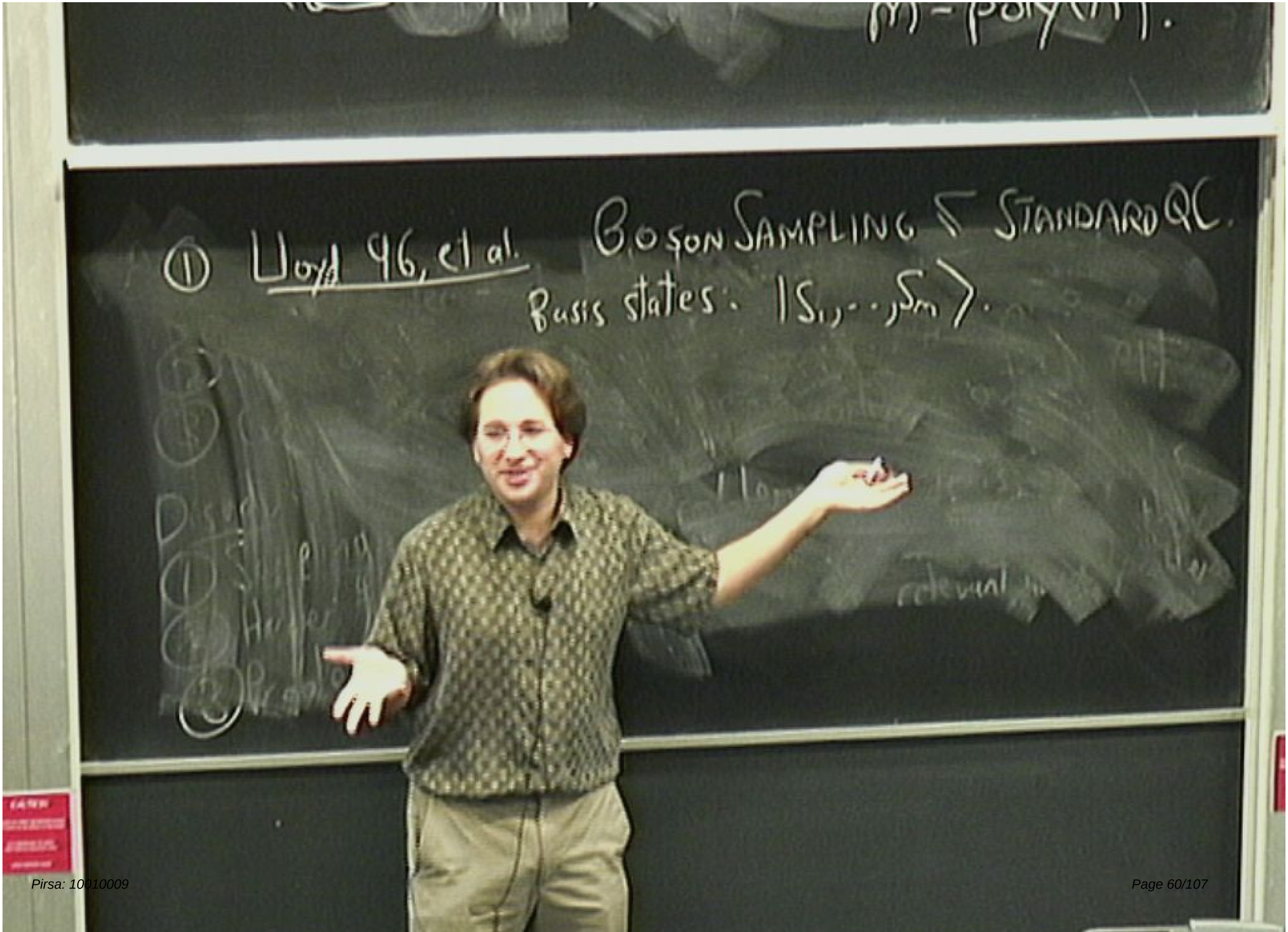
(2) Harder to convince ... "useful"?

(3) Problems not "useful"?


|S⟩ = |s1, ..., sm⟩ |T⟩ = |t1, ..., tm⟩
|S⟩ = |n, 0, 0, ..., 0⟩
|T⟩ = |n, 0, ..., 0⟩
Then, measure V|I⟩
in computational basis
& output result
A
B
m
Pr = |Per(B)|^2 / s1! ... sm!
V_{S,T} = U11^n n!
Per [U11 ... U11; U11 ... U11]
For us
m = poly(n).
U
relevant hard problem
order to convin
blems not "useful"?


(n) = poly(n).
① Lloyd 96, et al.


m = poly(n).
① Lloyd 96, et al.
BOSONSAMPLING ⊆ STANDARD QC.


m = poly(n).
① Lloyd 96, et al.
BosonSampling ⊆ StandardQC.
Basis states: |S1,...,Sm⟩.

$m = \mathrm{poly}(n)$.

(1) Lloyd 96, et al. BOSON SAMPLING $\leq$ STANDARD QC.

Basis states: $|S_1, \ldots, S_m\rangle$.

Fact: Every $m \times m$ unitary U can be decomposed as $U_1 \cdots U_T$, $T = O(m^2)$, where each U_i acts nontrivially on ≤ 2 modes.

$m = \mathrm{poly}(n)$.

① Lloyd 96, et al. BOSON SAMPLING $\leq$ STANDARD QC.

Basis states: $|S_1, \ldots, S_m\rangle$.

Fact: Every $m \times m$ unitary U can be decomposed as $U_1 \cdots U_T$, $T = O(m^2)$, where each U_i acts nontrivially on ≤ 2 modes.

② BOSON SAMPLING $\leq$ Log-depth quantum circuits

$m = \text{poly}(n)$.

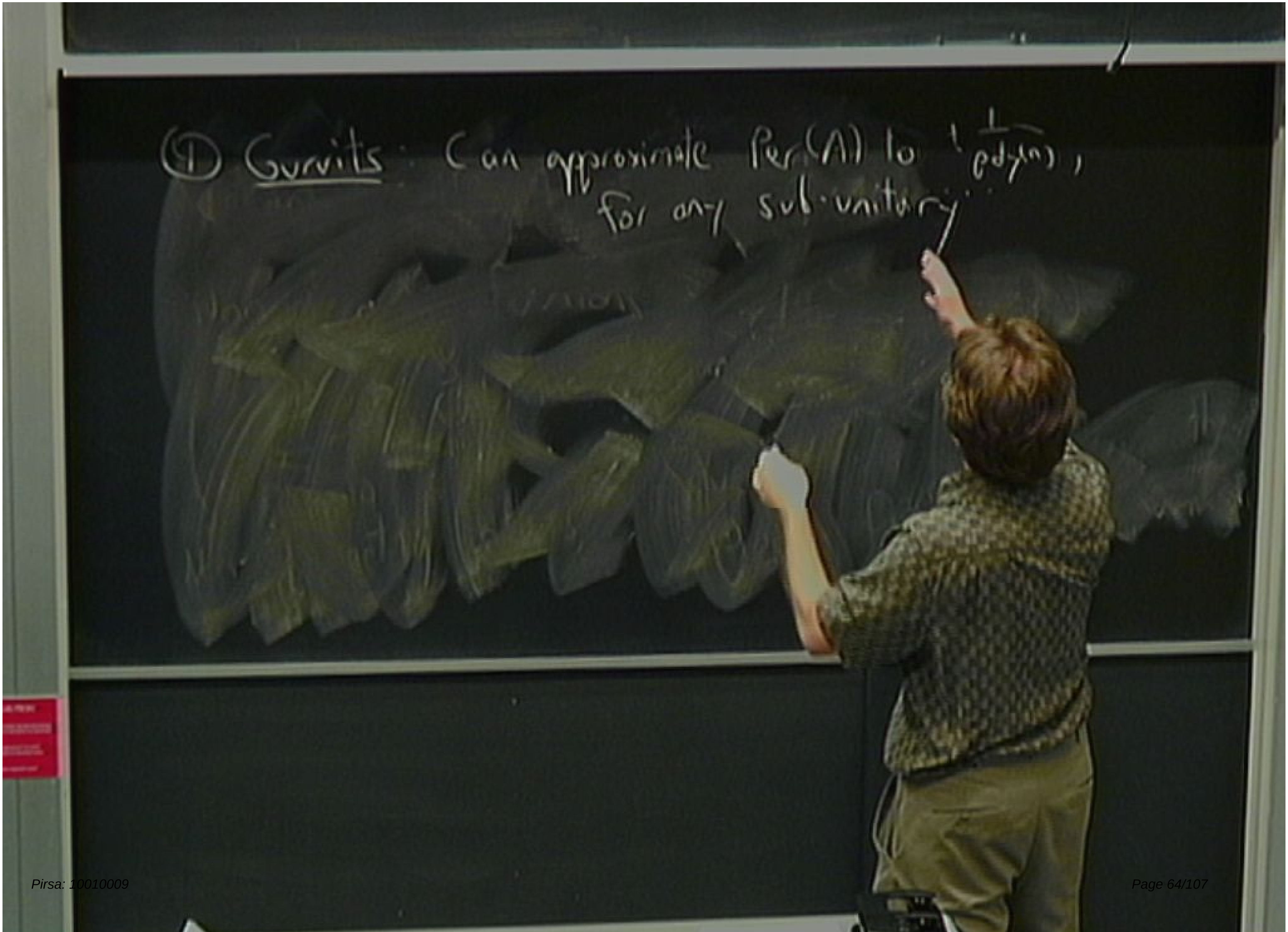
① Lloyd 96, et al. BOSON SAMPLING $\leq$ STANDARD QC.

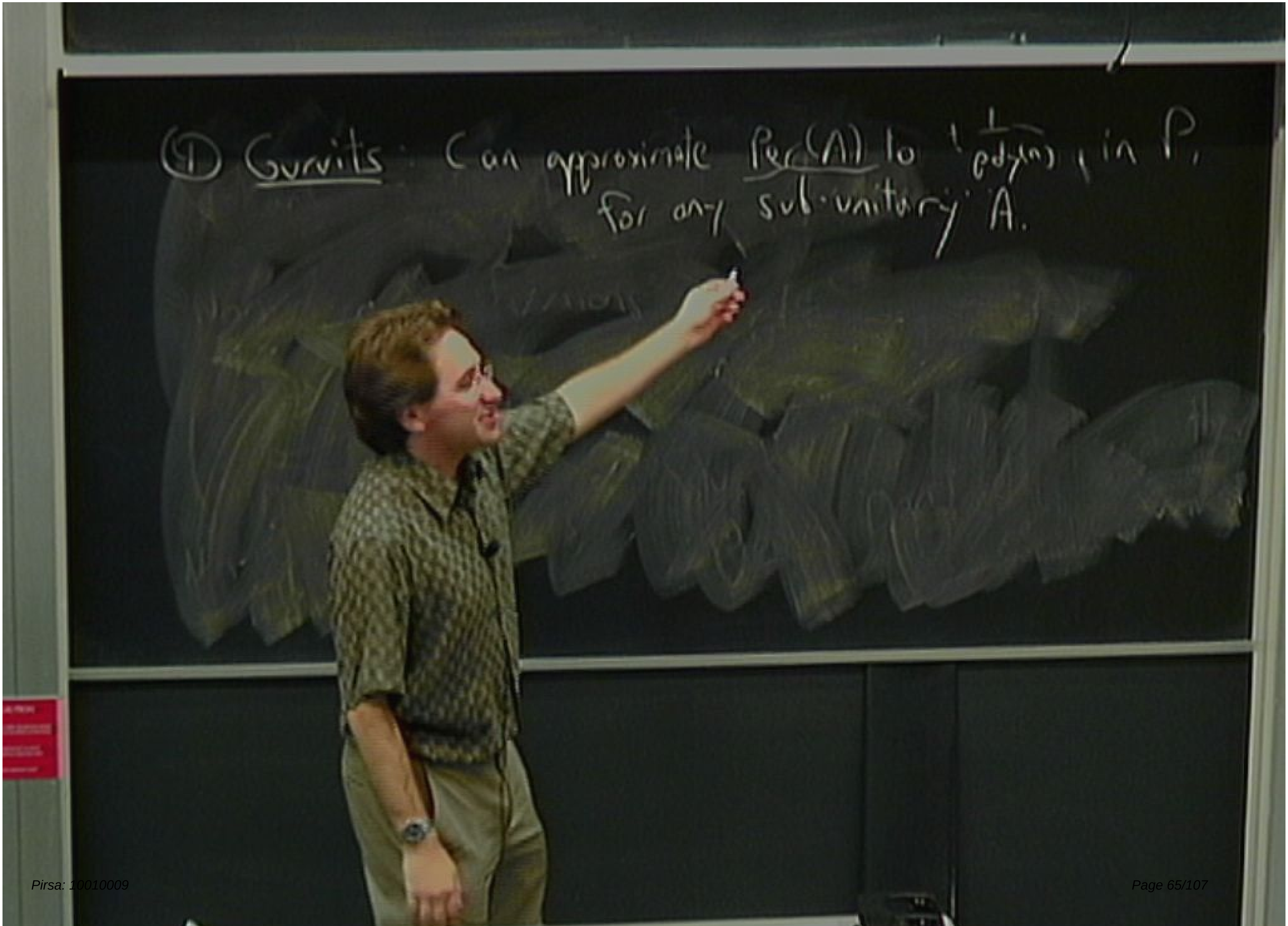
Basis states: $|S_1, \ldots, S_m\rangle$.

Fact: Every $m \times m$ unitary U can be decomposed as $U_1 \cdots U_T$, $T = O(m^2)$, where each U_i acts nontrivially on ≤ 2 modes.

② BOSON SAMPLING $\leq$ Log-depth quantum circuits

③ KLM: Add adaptive measurements $\Rightarrow$ Universal QC.

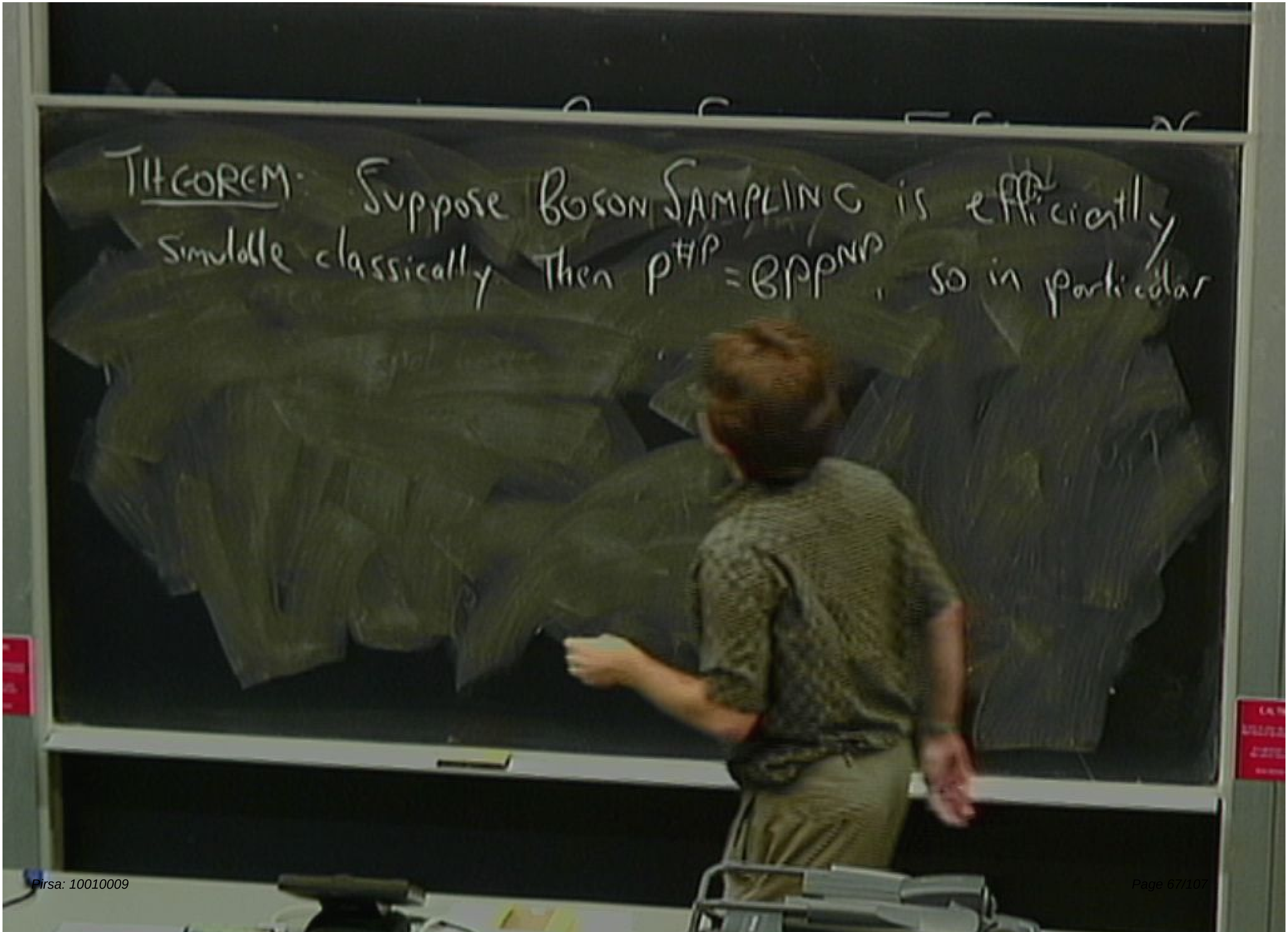

① Gurvits: Can approximate Per(A) to ± 1/poly(n),
for any sub-unitary


① Gurvits: Can approximate Per(A) to ± 1/poly(n) in P,
for any sub-unitary A.

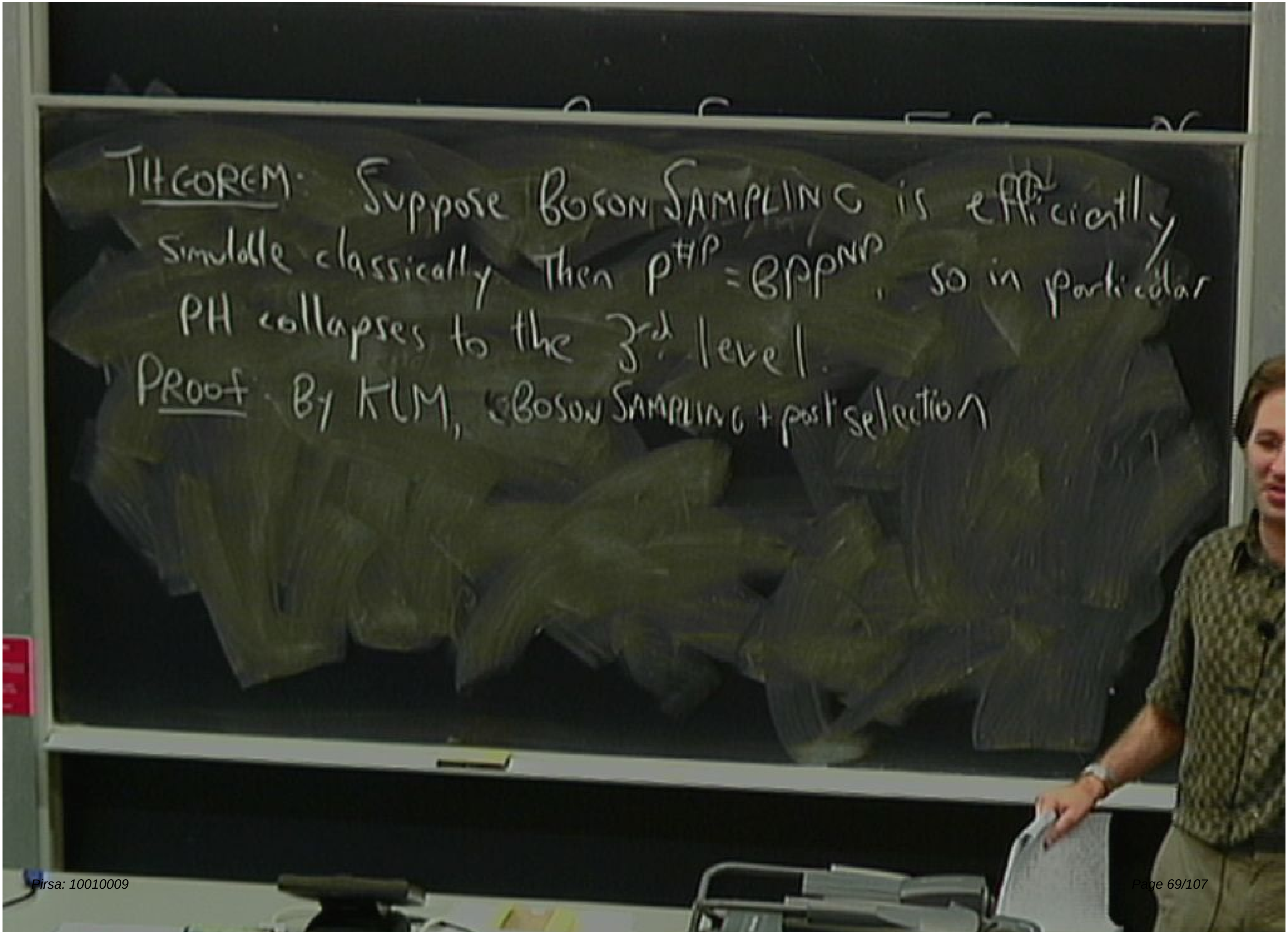
(4) <u>Gurvits</u>: Can approximate <u>Per(A)</u> to $\pm\frac{1}{\mathrm{poly}(n)}$ in P, for any sub-unitary A.

(5) <u>Bartlett, Sanders</u>: Can simulate efficiently classically if inputs are coherent states

$$|\alpha\rangle = e^{-|\alpha|^2/2} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle$$

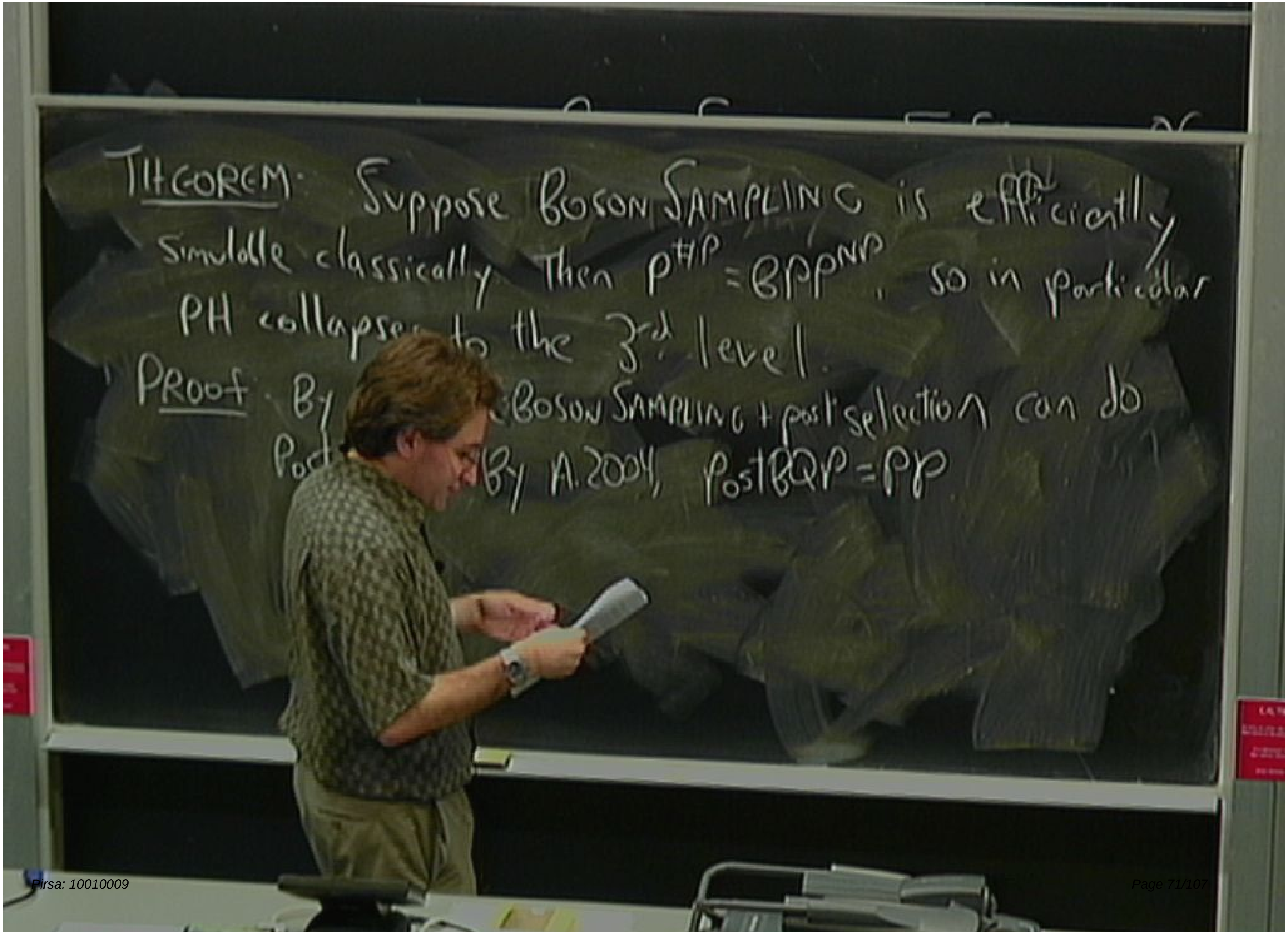

THEOREM: Suppose BOSON SAMPLING is efficiently
simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular

THEOREM: Suppose BOSON SAMPLING is efficiently simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular PH collapses to the 3rd level.


THEOREM: Suppose BOSON SAMPLING is efficiently
simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular
PH collapses to the 3rd level.
PROOF: By KLM, BOSON SAMPLING + postselection

THEOREM: Suppose BosonSampling is efficiently simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular PH collapses to the 3rd level.

Proof: By KLM, BosonSampling + postselection can do PostBQP


THEOREM: Suppose BOSON SAMPLING is efficiently
simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular
PH collapses to the 3rd level.
PROOF: By
BOSON SAMPLING + postselection can do
By A. 2004, PostBQP = PP

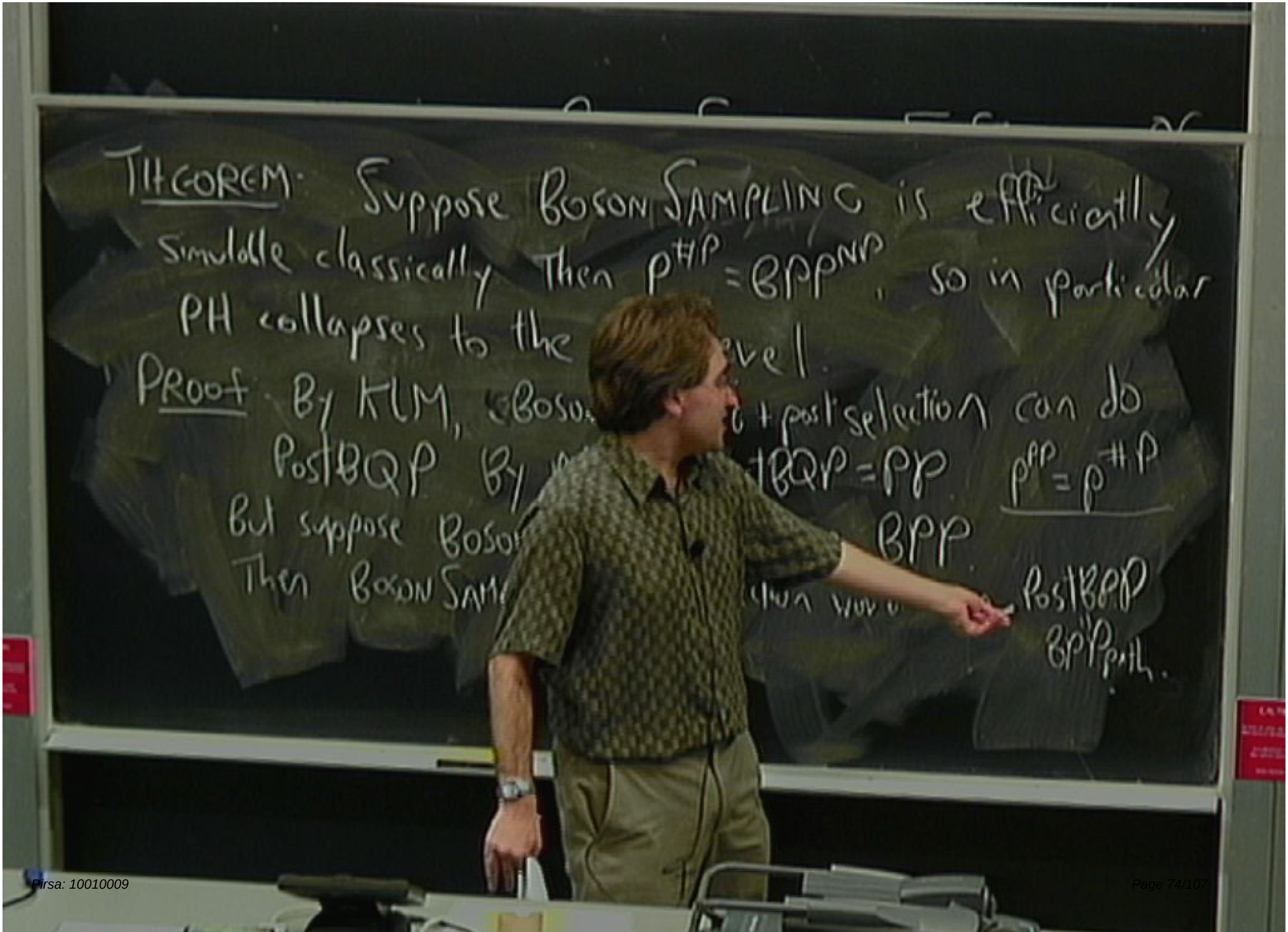
THEOREM: Suppose BosonSampling is efficiently simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular PH collapses to the 3rd level.

Proof: By KLM, BosonSampling + postselection can do PostBQP. By A. 2004, PostBQP = PP. $P^{PP} = P^{\#P}$

THEOREM: Suppose BosonSampling is efficiently simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular PH collapses to the 3rd level.

Proof: By KLM, BosonSampling + postselection can do PostBQP. By A. 2004, PostBQP = PP. $P^{PP} = P^{\#P}$

But suppose BosonSampling were in BPP.


THEOREM: Suppose BOSON SAMPLING is efficiently
simulable classically. Then P^#P = BPP^NP, so in particular
PH collapses to the
PROOF: By KLM,
+ postselection can do
PostBQP
PostBQP = PP
P^PP = P^#P
But suppose
BPP
Then BosonSam
PostBPP
BPP_path

THEOREM: Suppose BosonSampling is efficiently simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular PH collapses to the 3rd level.

Proof: By KLM, BosonSampling + postselection can do PostBQP. By A. 2004, PostBQP = PP. $P^{PP} = P^{\#P}$

But suppose BosonSampling were in BPP. Then BosonSampling + postselection would be in PostBPP $\subseteq$ BPP_{path}.

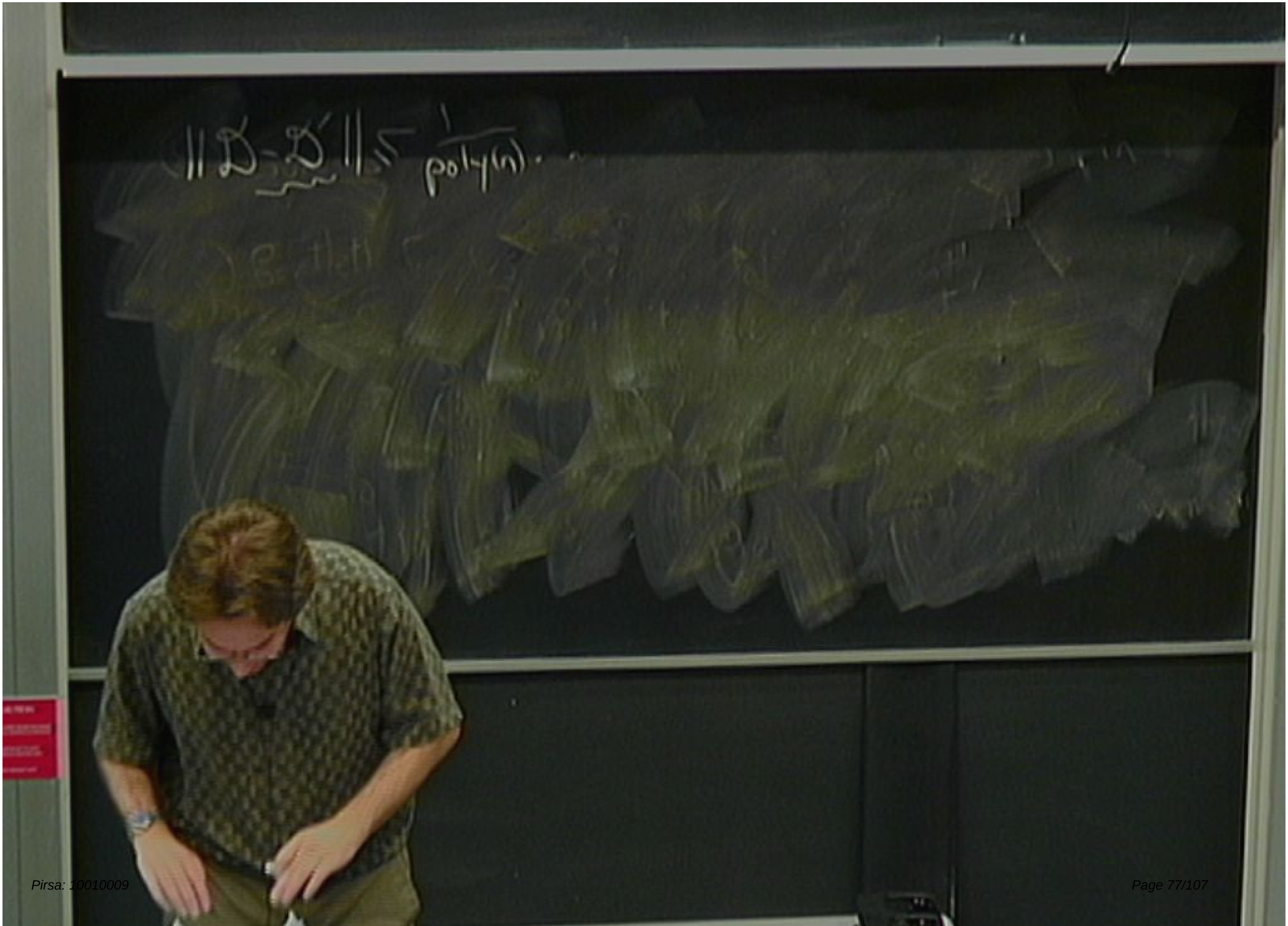
THEOREM: Suppose BOSON SAMPLING is efficiently simulable classically. Then $P^{\#P} = BPP^{NP}$, so in particular PH collapses to the 3rd level.

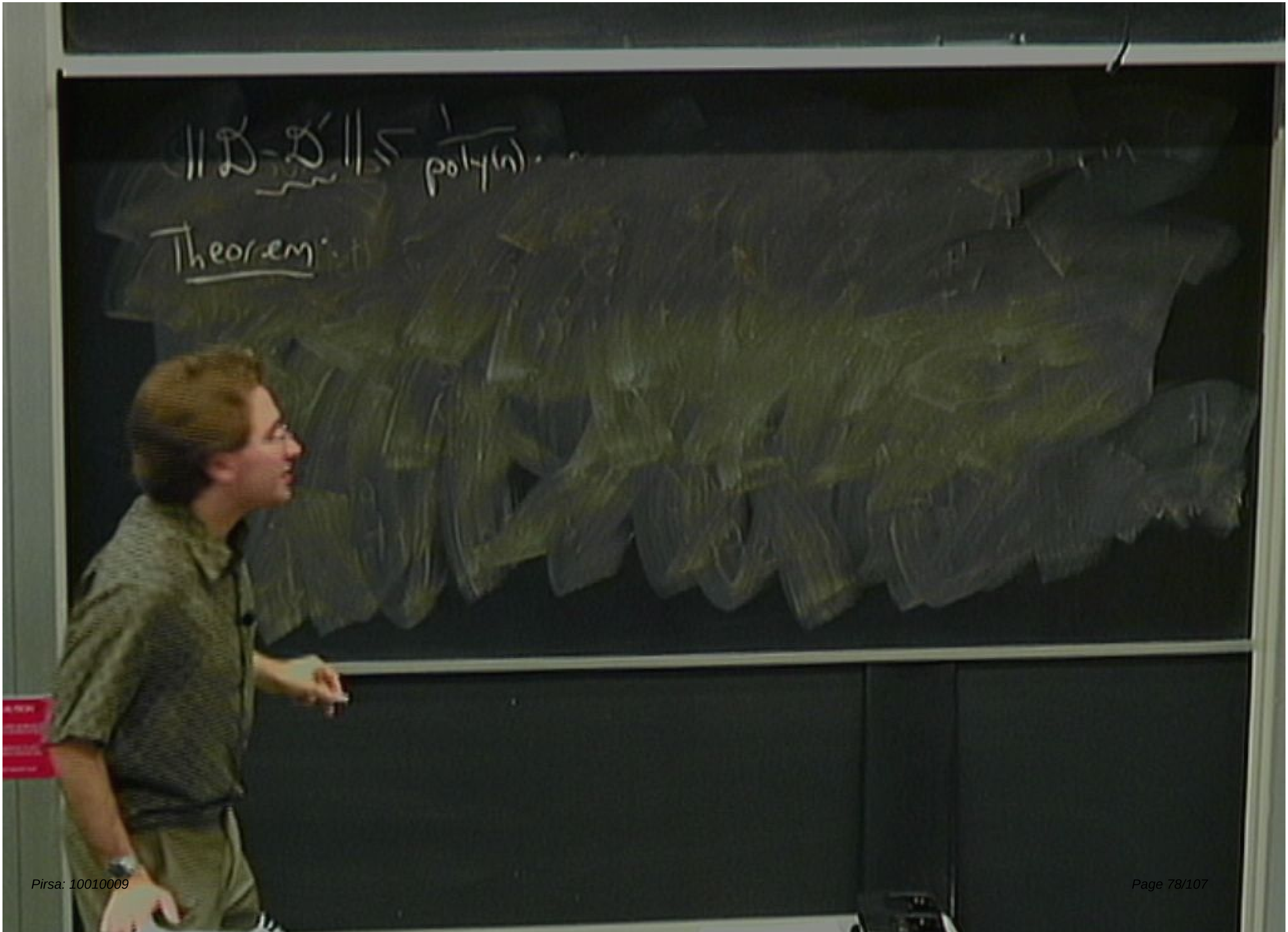
PROOF: By KLM, BOSON SAMPLING + postselection can do PostBQP. By A. 2004, PostBQP = PP. $P^{PP} = P^{\#P}$

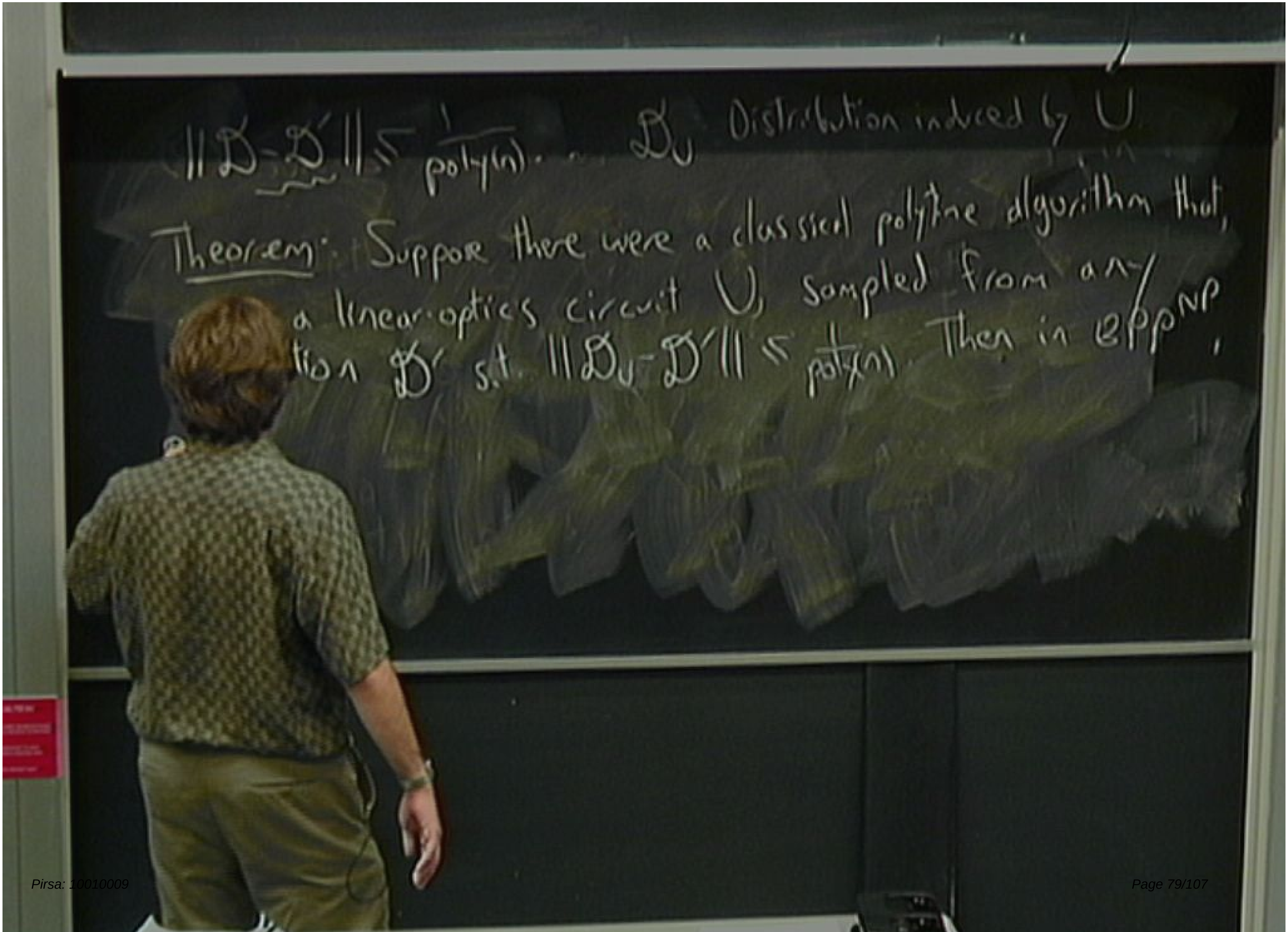
But suppose BOSON SAMPLING were in BPP.

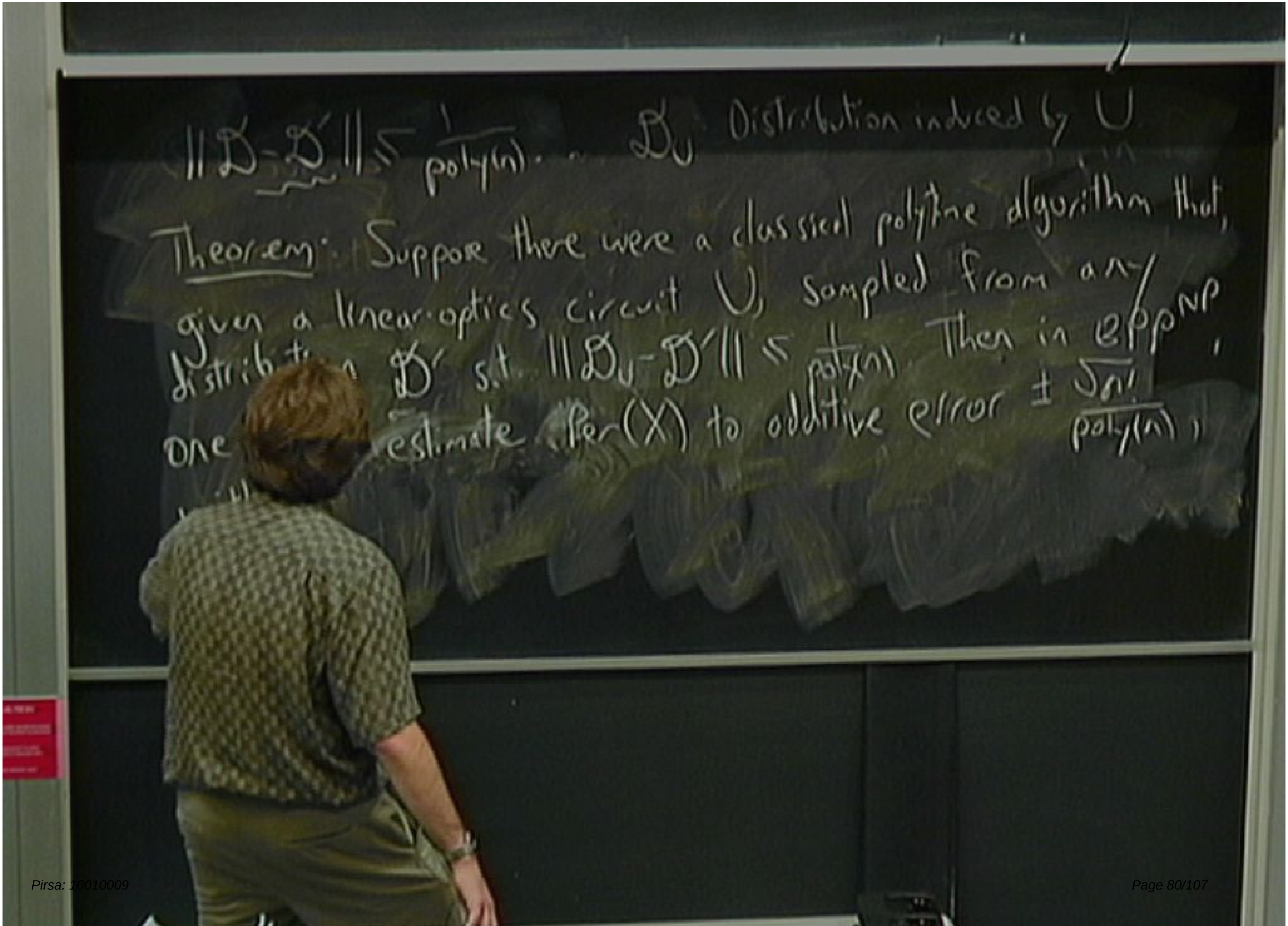
Then BOSON SAMPLING + postselection would be in PostBPP.

Hence $BPP_{path} = PP$, so $P^{\#P} = P^{PP} = P^{BPP_{path}} \subseteq PH$. BPP_{path}.


$\|\tilde{D}-\tilde{D}'\| \leq \frac{1}{\mathrm{poly}(n)}$


$\|\mathcal{D} - \mathcal{D}'\| \leq \frac{1}{\text{poly}(n)}$
Theorem:


||D - D'|| ≤ 1/poly(n).
D_U Distribution induced by U
Theorem: Suppose there were a classical polytime algorithm that,
a linear-optics circuit U, sampled from any
tion D' s.t. ||D_U - D'|| ≤ 1/poly(n). Then in BPP^NP.


$\|\mathcal{D}-\mathcal{D}'\| \leq \frac{1}{\text{poly}(n)}$.
$\mathcal{D}_U$: Distribution induced by U.
Theorem: Suppose there were a classical polytime algorithm that,
given a linear-optics circuit U, sampled from any
distrib... $\mathcal{D}'$ s.t. $\|\mathcal{D}_U-\mathcal{D}'\| \leq \frac{1}{\text{poly}(n)}$. Then in BPP^{NP},
one ... estimate Per(X) to additive error $\pm \frac{\sqrt{n!}}{\text{poly}(n)}$,

$\|\mathcal{D}-\mathcal{D}'\| \le \frac{1}{\mathrm{poly}(n)}$ $\mathcal{D}_U$: Distribution induced by U

<u>Theorem</u>: Suppose there were a classical polytime algorithm that, given a linear-optics circuit U, sampled from any distribution $\mathcal{D}'$ s.t. $\|\mathcal{D}_U-\mathcal{D}'\| \le \frac{1}{\mathrm{poly}(n)}$. Then in BPP^{NP}, one could estimate $\mathrm{Per}(X)$ to additive error $\pm \frac{\sqrt{n!}}{\mathrm{poly}(n)}$, with $1-\frac{1}{\mathrm{poly}(n)}$ over X drawn from a distribution $\mathcal{G}$ of i.i.d. $\mathcal{N}(0,1)$ Gaussian entries.

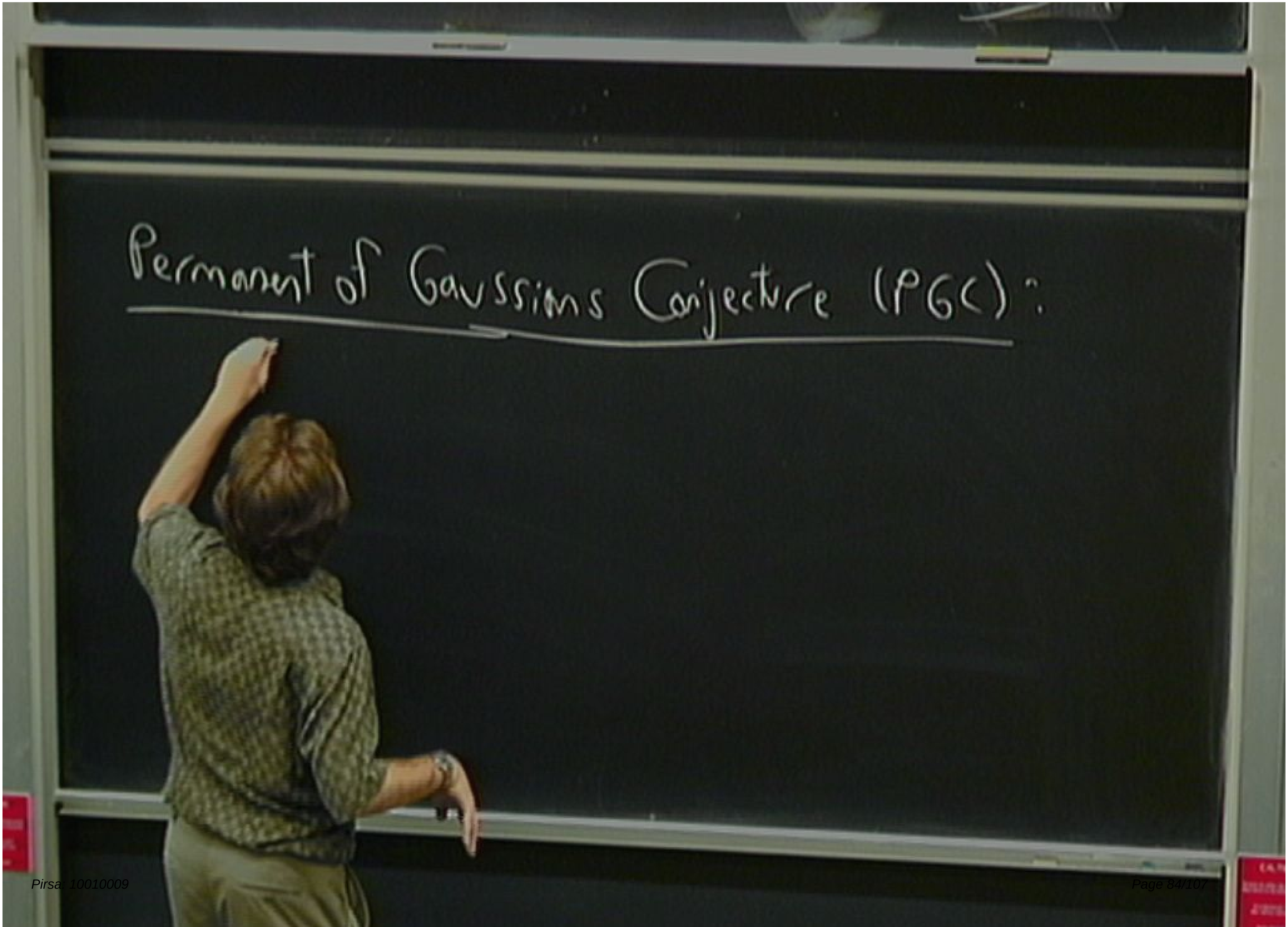
$\|\mathcal{D}-\mathcal{D}'\| \leq \frac{1}{\text{poly}(n)}$. $\mathcal{D}_U$: Distribution induced by U

Theorem: Suppose there were a classical polytime algorithm that, given a linear-optics circuit U, sampled from any distribution $\mathcal{D}'$ s.t. $\|\mathcal{D}_U - \mathcal{D}'\| \leq \frac{1}{\text{poly}(n)}$. Then in BPP^{NP}, one could estimate $\text{Per}(X)$ to additive error $\pm \frac{\sqrt{n!}}{\text{poly}(n)}$, with $1-\frac{1}{\text{poly}(n)}$ over X drawn from a distribution $\mathcal{G}$ of i.i.d. $\mathcal{N}(0,1)$ Gaussian entries.

$\|\mathcal{D}-\mathcal{D}'\| \le \frac{1}{\mathrm{poly}(n)}$. $\mathcal{D}_U$ Distribution induced by U

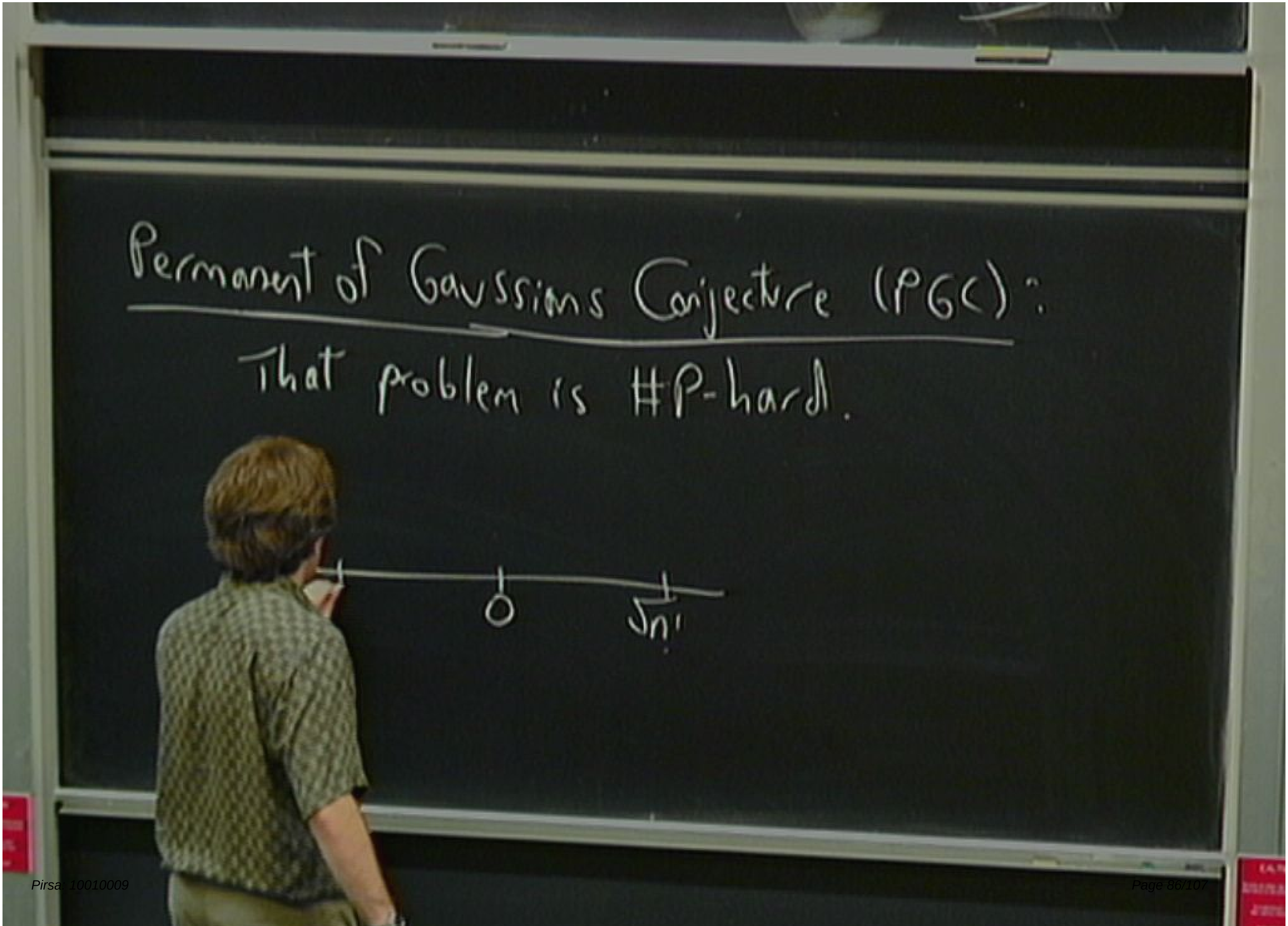
<u>Theorem</u>: Suppose there were a classical polytime algorithm that, given a linear-optics circuit U, sampled from any distribution $\mathcal{D}'$ s.t. $\|\mathcal{D}_U-\mathcal{D}'\| \le \frac{1}{\mathrm{poly}(n)}$. Then in $\mathsf{BPP}^{\mathsf{NP}}$, one could estimate $\mathrm{Per}(X)$ to additive error $\pm \frac{\sqrt{n!}}{\mathrm{poly}(n)}$, with $1-\frac{1}{\mathrm{poly}(n)}$ probability over X drawn from a distribution $\mathcal{G}$ of i.i.d. $\mathcal{N}(0,1)$ Gaussian entries.

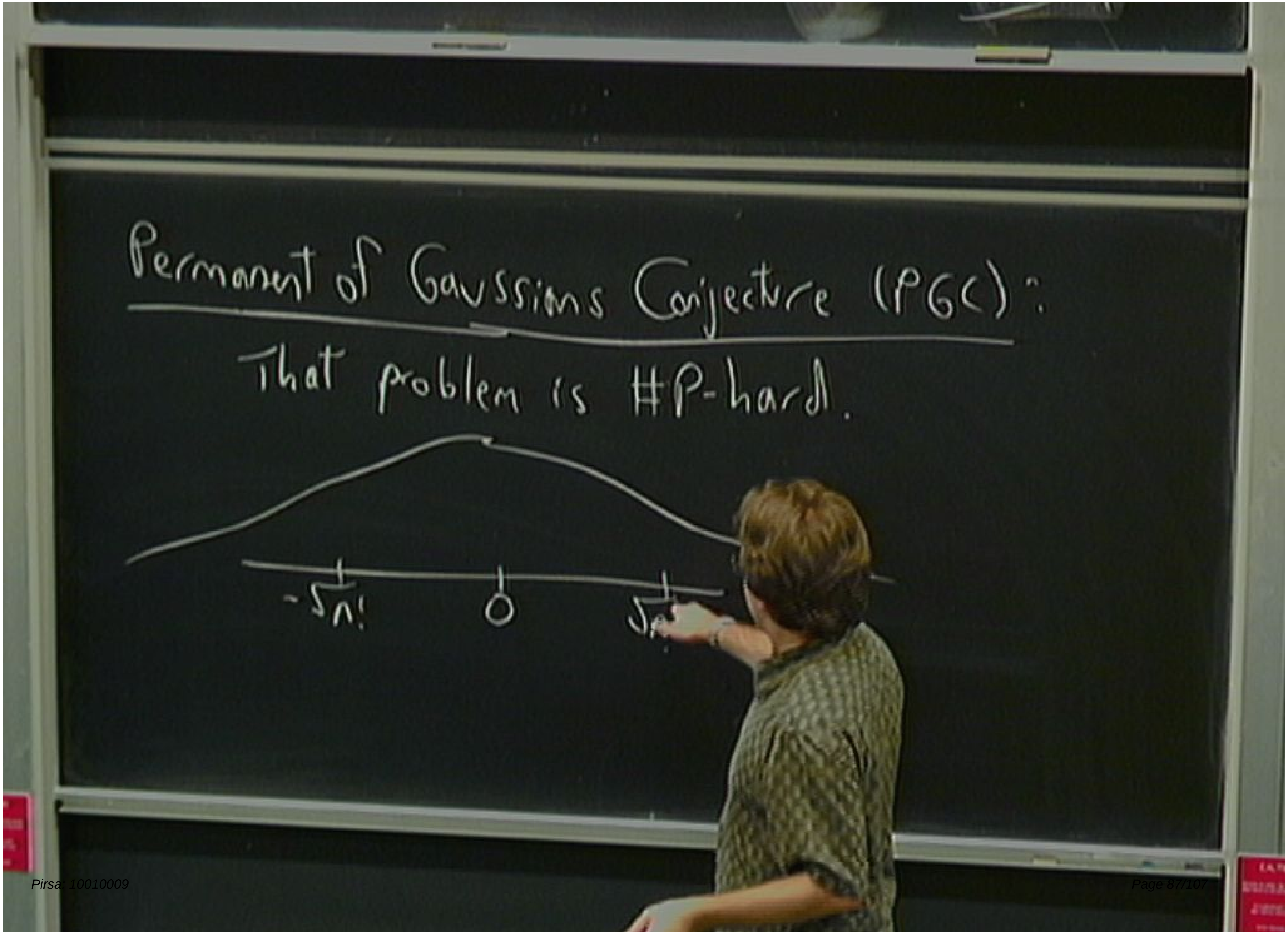
$\mathrm{Per}(X) \sim \sqrt{n!}$


Permanent of Gaussians Conjecture (PGC):

Permanent of Gaussians Conjecture (PGC):

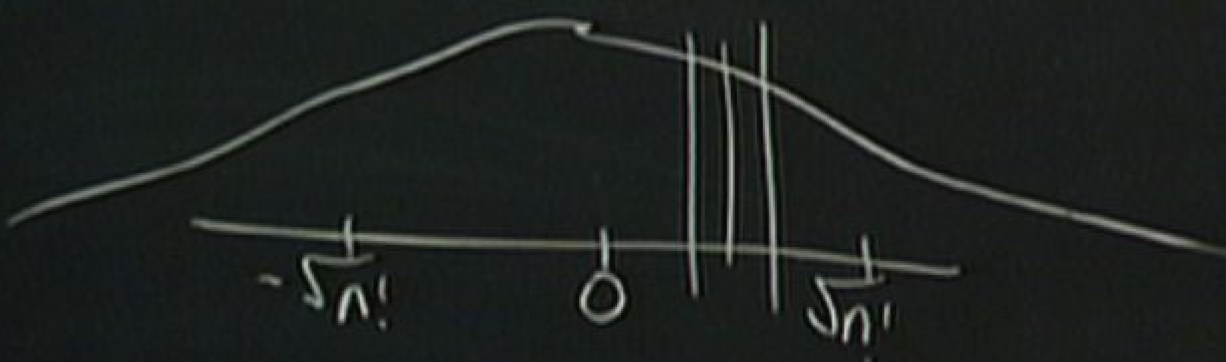
That problem is #P-hard.


Permanent of Gaussians Conjecture (PGC):
That problem is #P-hard.
0
$\sqrt{n!}$


Permanent of Gaussians Conjecture (PGC):
That problem is #P-hard.
$-\sqrt{n!}$
0

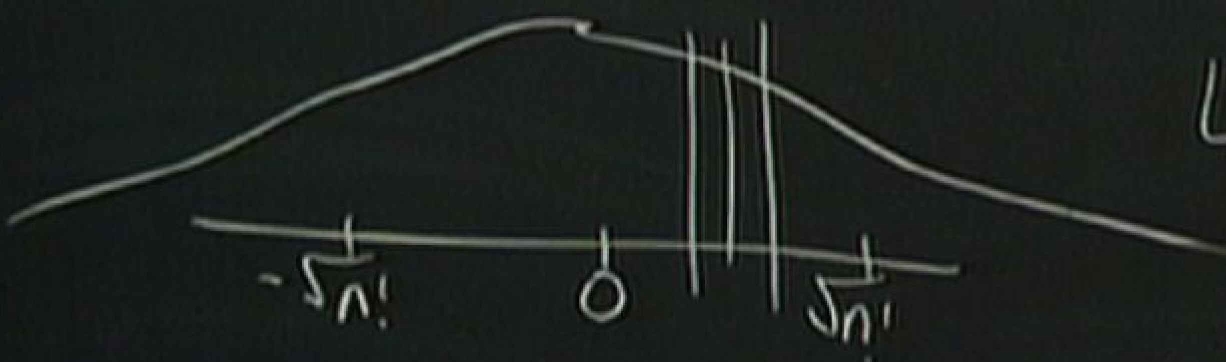
Permanent of Gaussians Conjecture (PGC):

That problem is #P-hard.



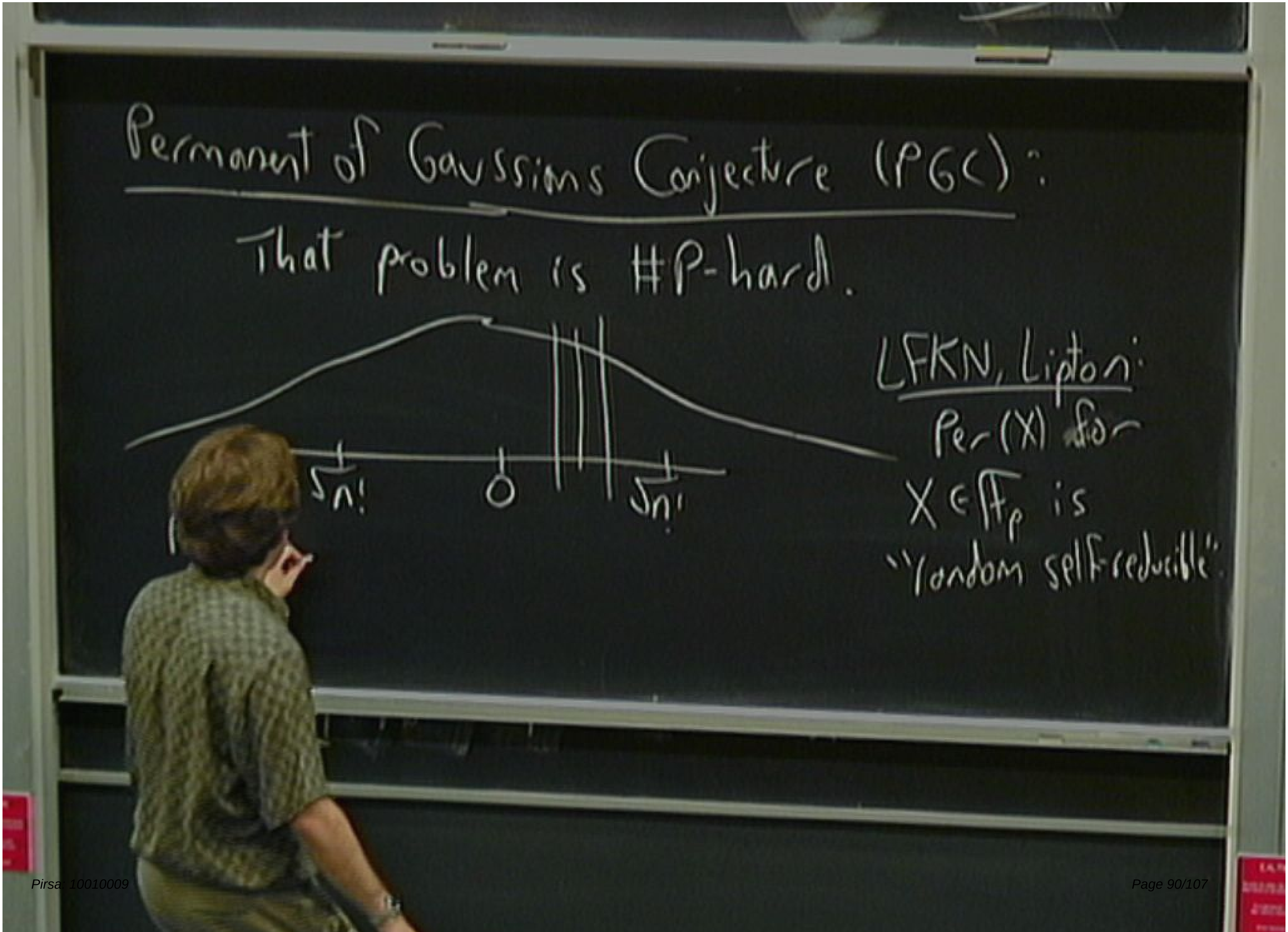
Permanent of Gaussians Conjecture (PGC):

That problem is #P-hard.



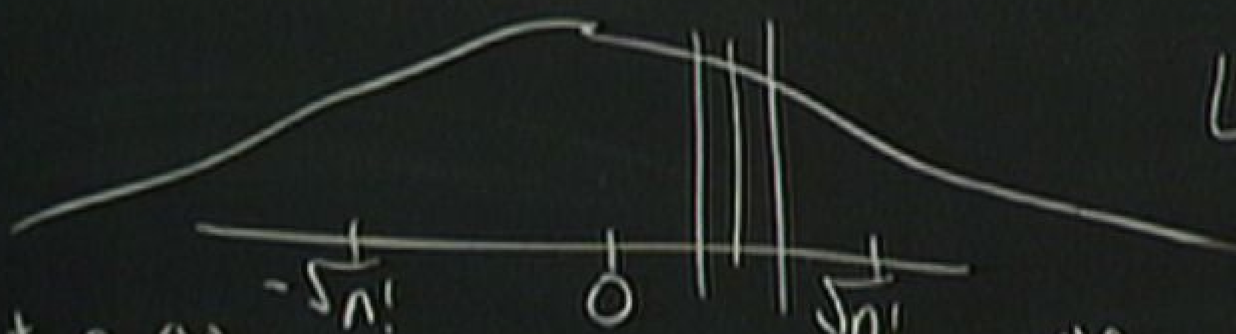
LFKN, Lipton:

$\mathrm{Per}(X)$ for $X \in \mathbb{F}_p$ is "random self-reducible".


Permanent of Gaussians Conjecture (PGC):
That problem is #P-hard.
√n!
0
√n!
LFKN, Lipton:
Per(X) for
X ∈ F_p is
"random self-reducible".

Permanent of Gaussians Conjecture (PGC):

That problem is #P-hard.



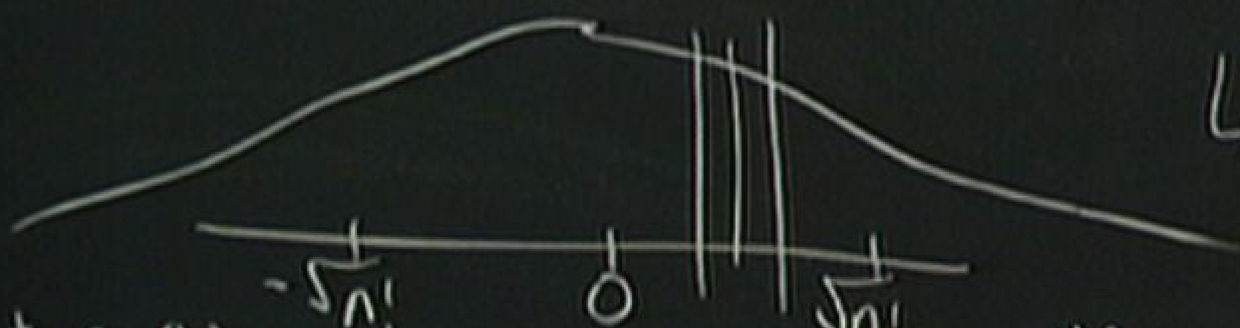
LFKN, Lipton:

$Per(X)$ for $X \in \mathbb{F}_p$ is "random self-reducible".

...ant $Per(X)$. Choose a random $Y \in \mathbb{F}_p^{n \times n}$.

Permanent of Gaussians Conjecture (PGC):

That problem is #P-hard.



LFKN, Lipton:

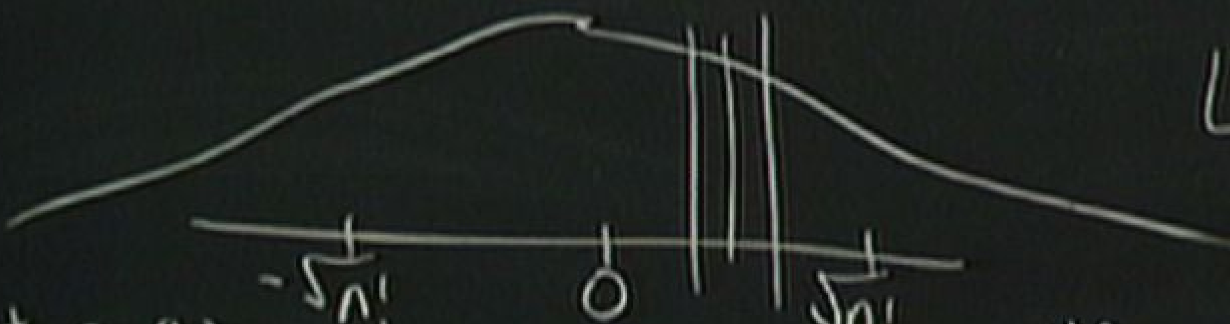
Per(X) for $X \in \mathbb{F}_p$ is "random self-reducible".

Want Per(X). Choose a random $Y \in \mathbb{F}_p^{n \times n}$.

$p(t) = \mathrm{Per}(X + tY)$. $p(0) = \mathrm{Per}(X)$.

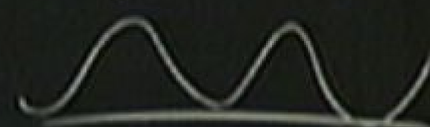
Permanent of Gaussians Conjecture (PGC):

That problem is #P-hard.



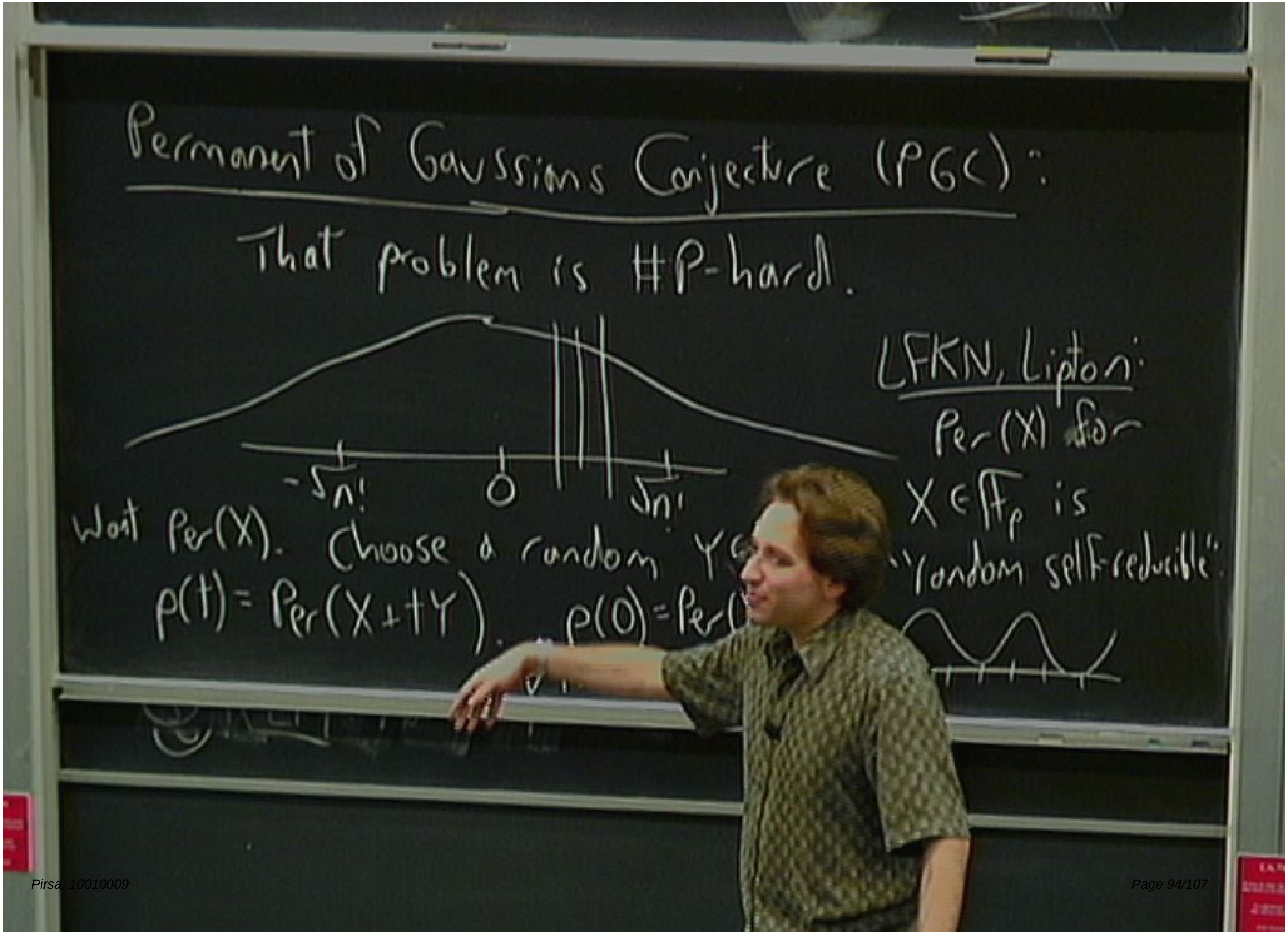
LFKN, Lipton:

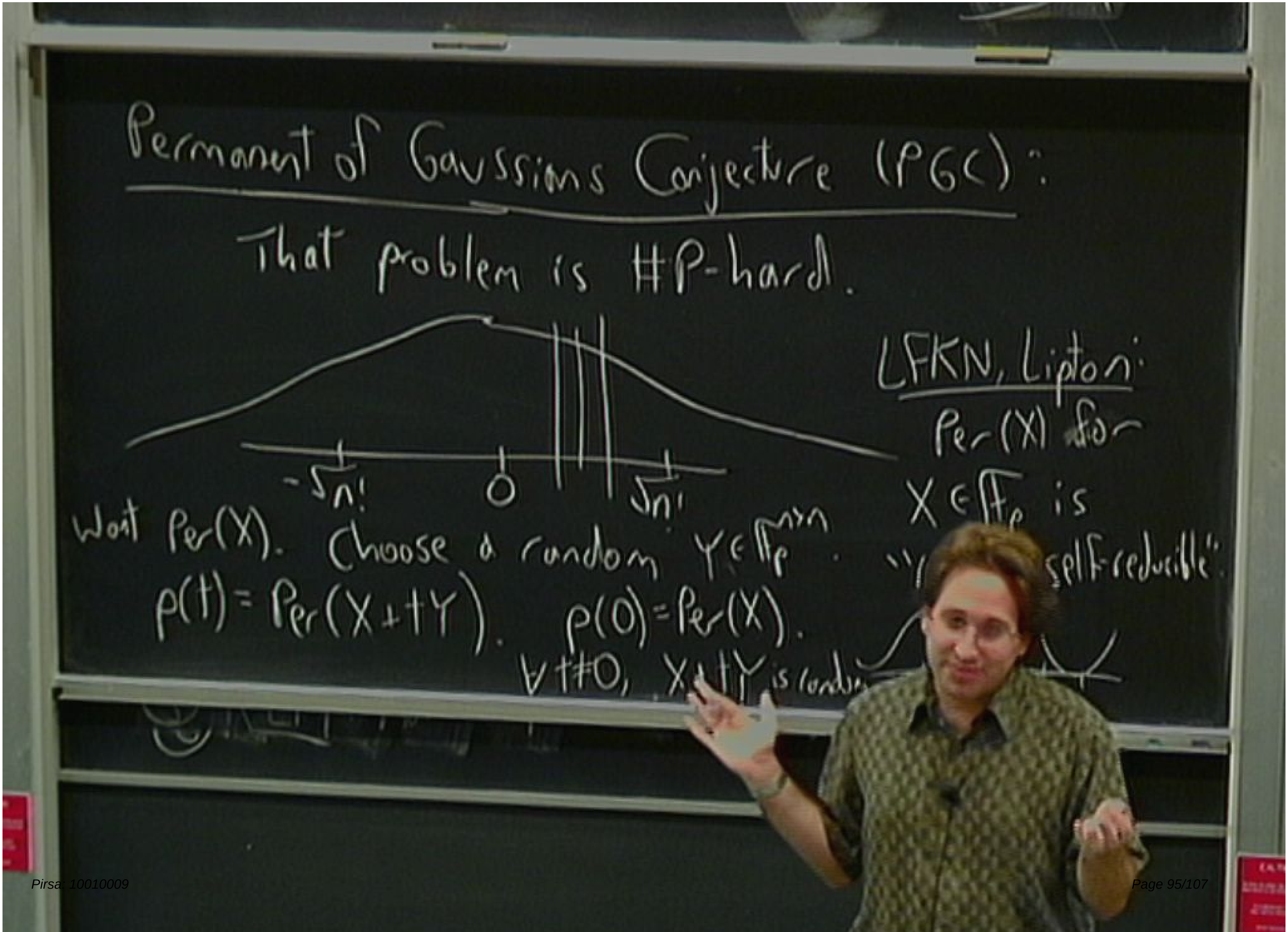
$\mathrm{Per}(X)$ for $X \in \mathbb{F}_p$ is "random self-reducible".



Want $\mathrm{Per}(X)$. Choose a random $Y \in \mathbb{F}_p^{n \times n}$.

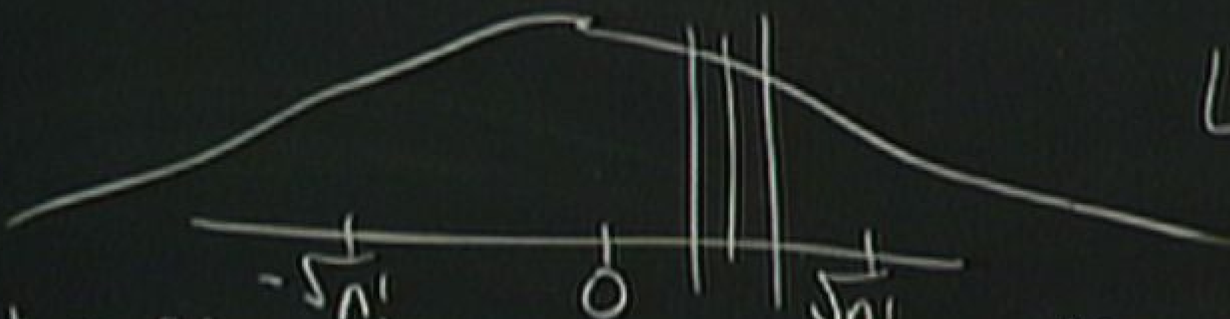
$p(t) = \mathrm{Per}(X + tY)$. $p(0) = \mathrm{Per}(X)$.


Permanent of Gaussians Conjecture (PGC):
That problem is #P-hard.
-√n!
0
√n!
LFKN, Lipton:
Per(X) for
X ∈ F_p is
"random self-reducible".
Want Per(X). Choose a random Y
p(t) = Per(X + tY).
p(0) = Per(


Permanent of Gaussians Conjecture (PGC):
That problem is #P-hard.
-√n!
0
√n!
LFKN, Lipton:
Per(X) for
X ∈ F_p is
self-reducible".
Want Per(X). Choose a random Y ∈ F_p^{n×n}.
p(t) = Per(X+tY). p(0)=Per(X).
∀t≠0, X+tY is random

Permanent of Gaussians Conjecture (PGC):

That problem is #P-hard.



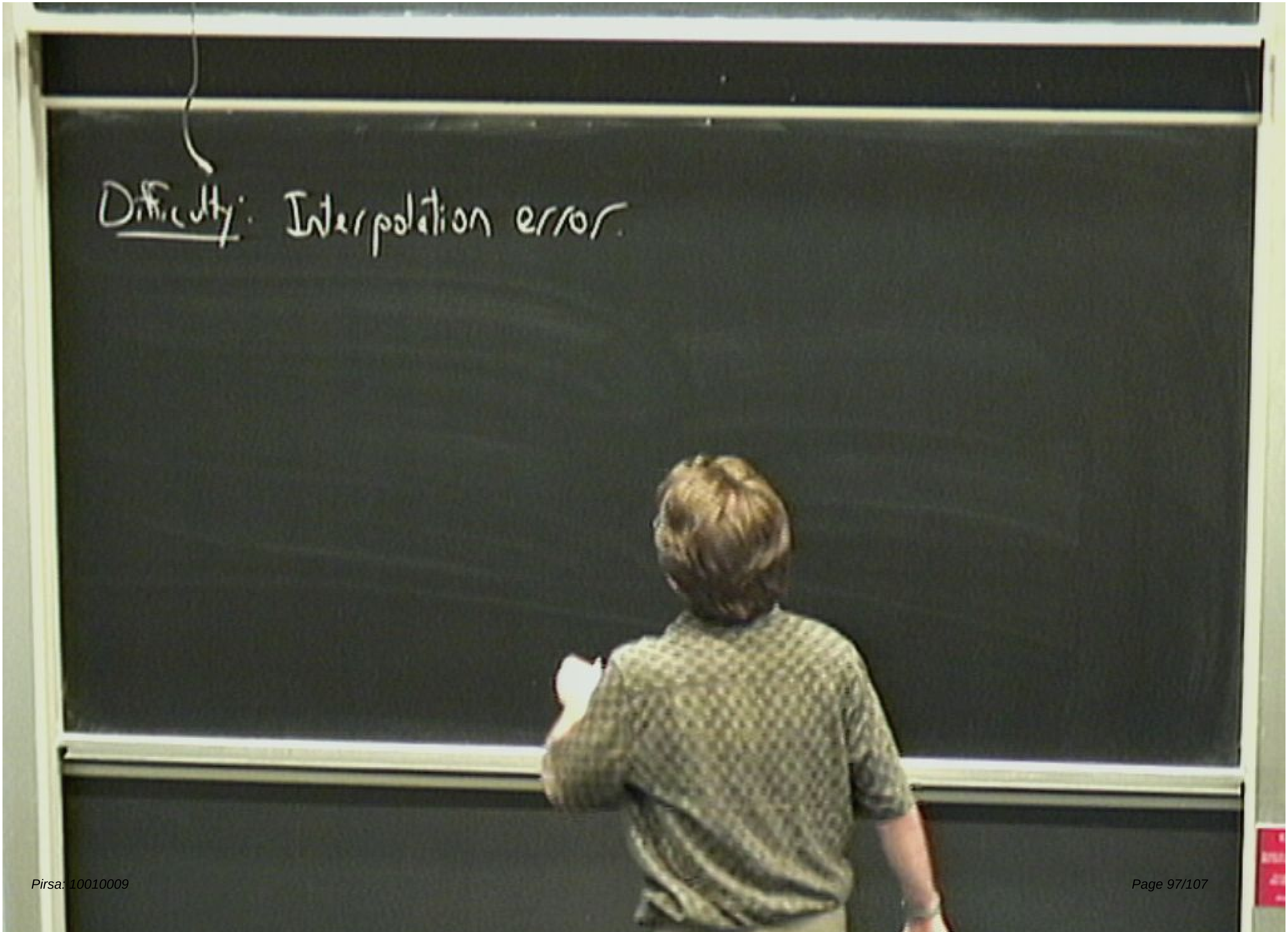
LFKN, Lipton:

$\mathrm{Per}(X)$ for $X \in \mathbb{F}_p$ is "random self-reducible".

Want $\mathrm{Per}(X)$. Choose a random $Y \in \mathbb{F}_p^{n \times n}$.

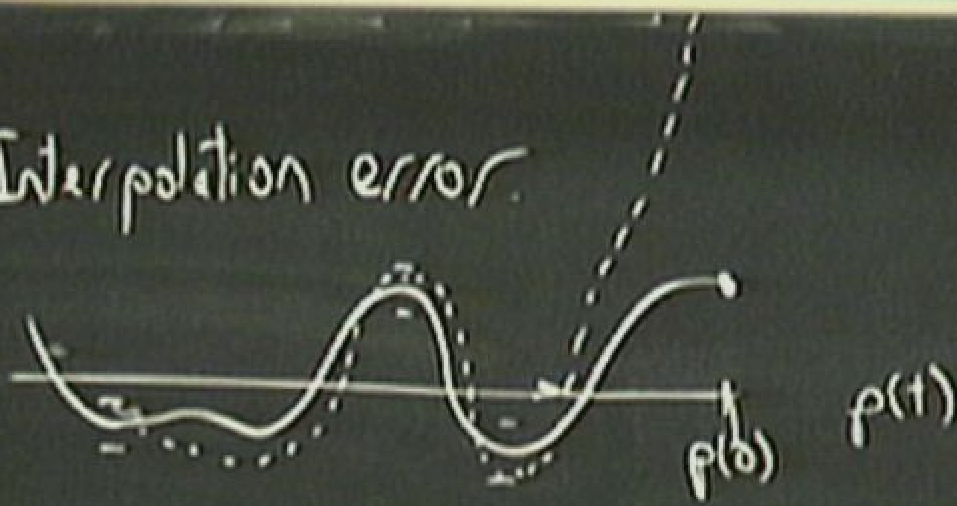
$p(t) = \mathrm{Per}(X + tY)$. $p(0) = \mathrm{Per}(X)$.

$\forall t \neq 0$, $X + tY$ is random


Difficulty: Interpolation error.

Difficulty: Interpolation error.

Difficulty: Interpolation error.



$\|\mathcal{D}-\mathcal{D}'\| \le \frac{1}{\text{poly}(n)}$. $\mathcal{D}_U$: Distribution induced by U

Theorem: Suppose there were a classical polytime algorithm that, given a linear-optics circuit U, sampled from any distribution $\mathcal{D}'$ s.t. $\|\mathcal{D}_U-\mathcal{D}'\| \le \frac{1}{\text{poly}(n)}$. Then in BPP^{NP}, one could estimate $\text{Per}(X)$ to additive error $\pm \frac{\sqrt{n!}}{\text{poly}(n)}$, with $1-\frac{1}{\text{poly}(n)}$ probability over X drawn from a distribution $\mathcal{G}$ of i.i.d. $\mathcal{N}(0,1)$ Gaussian entries.

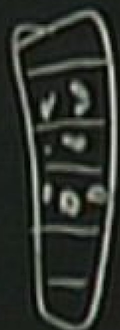
$\text{Per}(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant in P $\quad \mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\quad \mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

#P-complete.



Toda: $PH \subseteq P^{\#P}$.

$\mathrm{Per}(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant in P. $\quad \mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\quad \mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

#P-complete.

Toda: $PH \subseteq P^{\#P}$.

$\mathrm{Per}(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P)

$$\mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$$

Bosons $\longleftrightarrow$ Permanent

$\#P$-complete.

$$\mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$$

Toda: $PH \subseteq P^{\#P}$

"Bosonic Birthday Paradox"

$\mathrm{Per}(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P) $\quad Det(A) = \sum_{\sigma \in S_n} (-1)^{sgn(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\quad Per(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

#P-complete.

Toda: $PH \subseteq P^{\#P}$

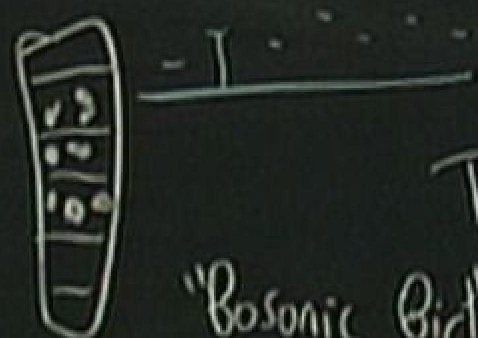
"Bosonic Birthday Paradox" $\quad$ Bosons: $m \sim \underline{2n^2}$.

of i.i.d. $N(0,1)$ Gaussian ... $\quad Per(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P) $\mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$



#P-complete.

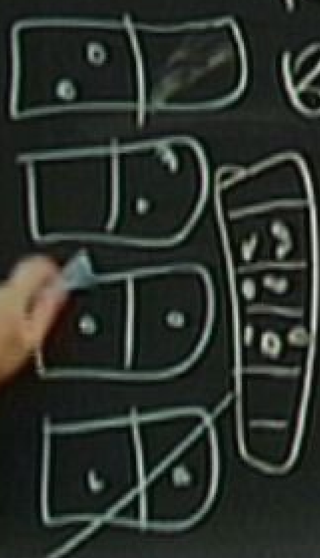
Toda: $PH \subseteq P^{\#P}$

"Bosonic Birthday Paradox"   Bosons: $m \sim 2n^2$.

of i.i.d. $N(0,1)$ ... $\mathrm{Per}(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P) $\quad \mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$



Bosons $\longleftrightarrow$ Permanent $\quad \mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$

$\#P$-complete.

Toda: $PH \subseteq P^{\#P}$

"Bosonic Birthday Paradox" $\quad$ Bosons: $m \sim 2n^2$.

of i.i.d. $N(0,1)$ ... $\quad \mathrm{Per}(X) \sim \sqrt{n!}$

Linear Optics: A way to bring the Permanent function into quantum computing

Fermions $\longleftrightarrow$ Determinant (in P) $\mathrm{Det}(A) = \sum_{\sigma \in S_n} (-1)^{\mathrm{sgn}(\sigma)} \prod_{i=1}^{n} a_{i\sigma(i)}$

Bosons $\longleftrightarrow$ Permanent $\mathrm{Per}(A) = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)}$.

#P-complete.

Toda: $PH \subseteq P^{\#P}$.

"Bosonic Birthday Paradox" Bosons: $m \sim 2n^2$.

of i.i.d. $N(0,1)$ $\mathrm{Per}(X) \sim \sqrt{n!}$