Infinite Randomness Expansion

with a constant number of devices

Matthew Coudron, **Henry Yuen**

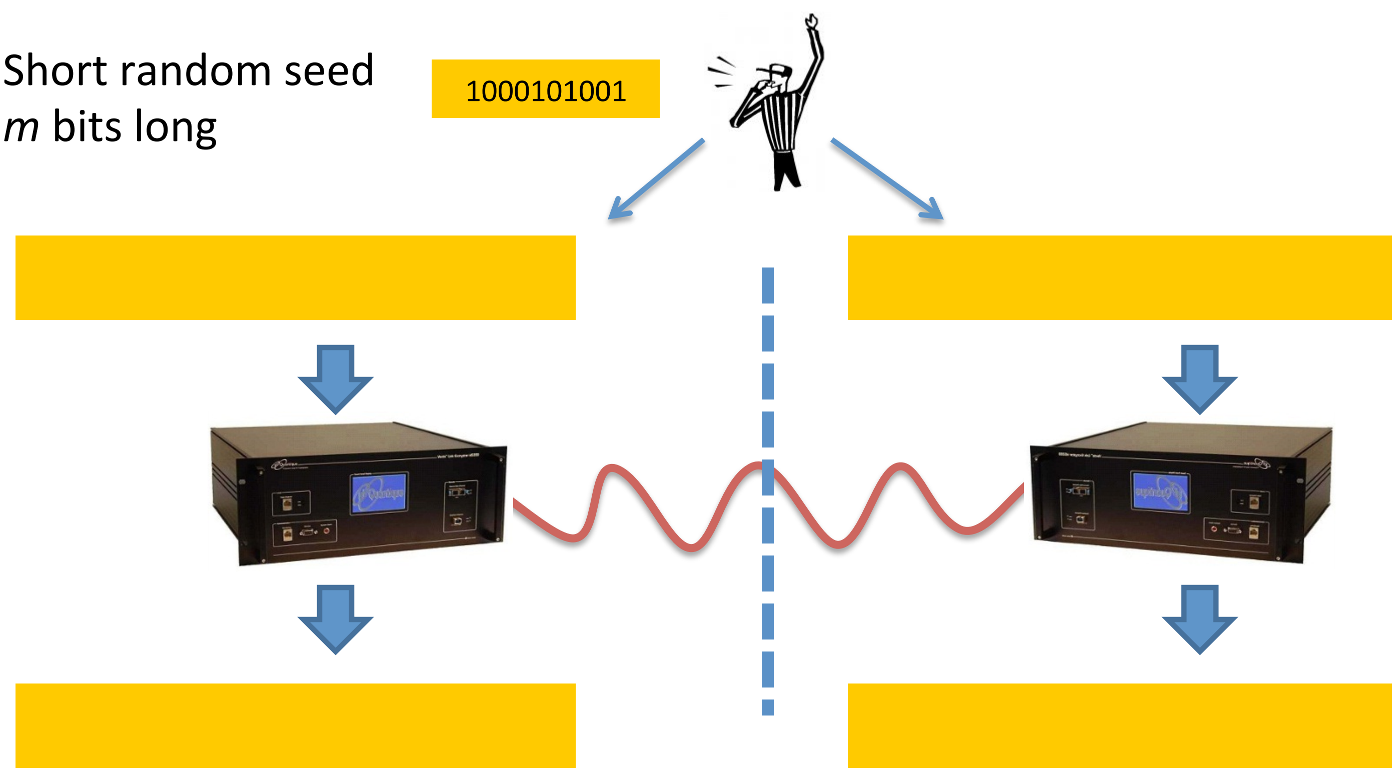
MIT CSAIL

February 26, 2014
Simons Institute

Randomness expansion protocols

Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

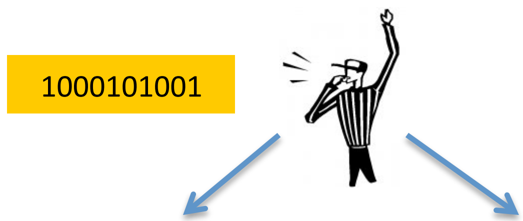
Short random seed
m bits long



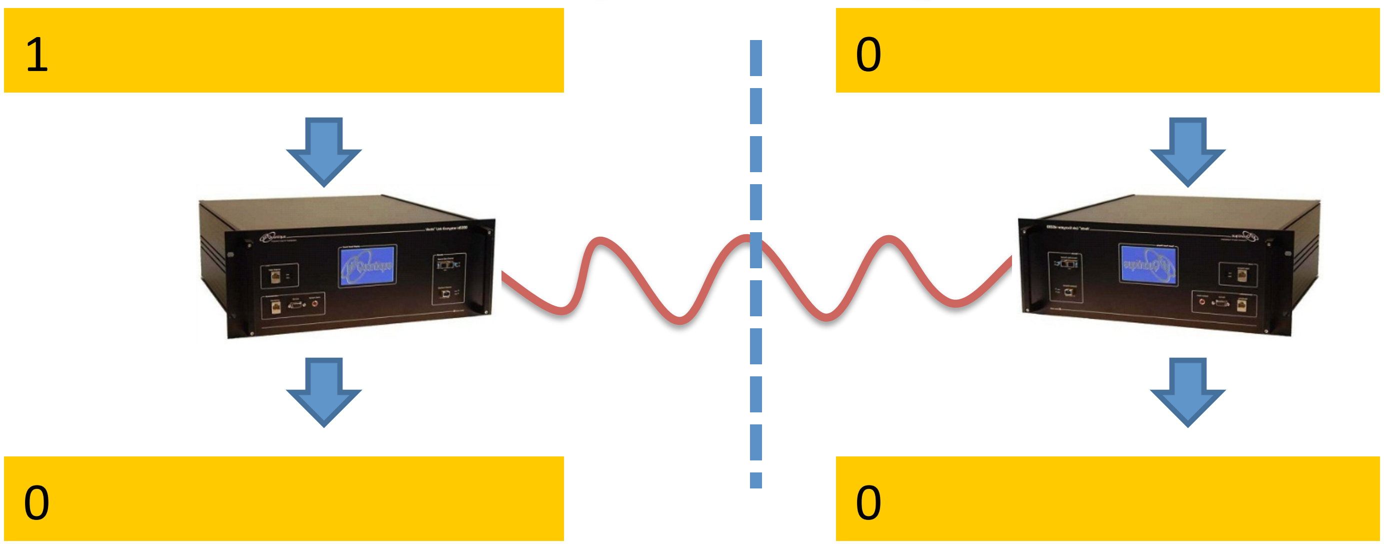
Randomness expansion protocols

Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

Short random seed
m bits long



n-bit inputs to devices are
deterministic function of seed S



Referee gives *n* bits to devices sequentially, and
collects *n* bits of output sequentially

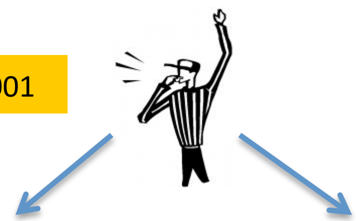
$n >> m$

Randomness expansion protocols

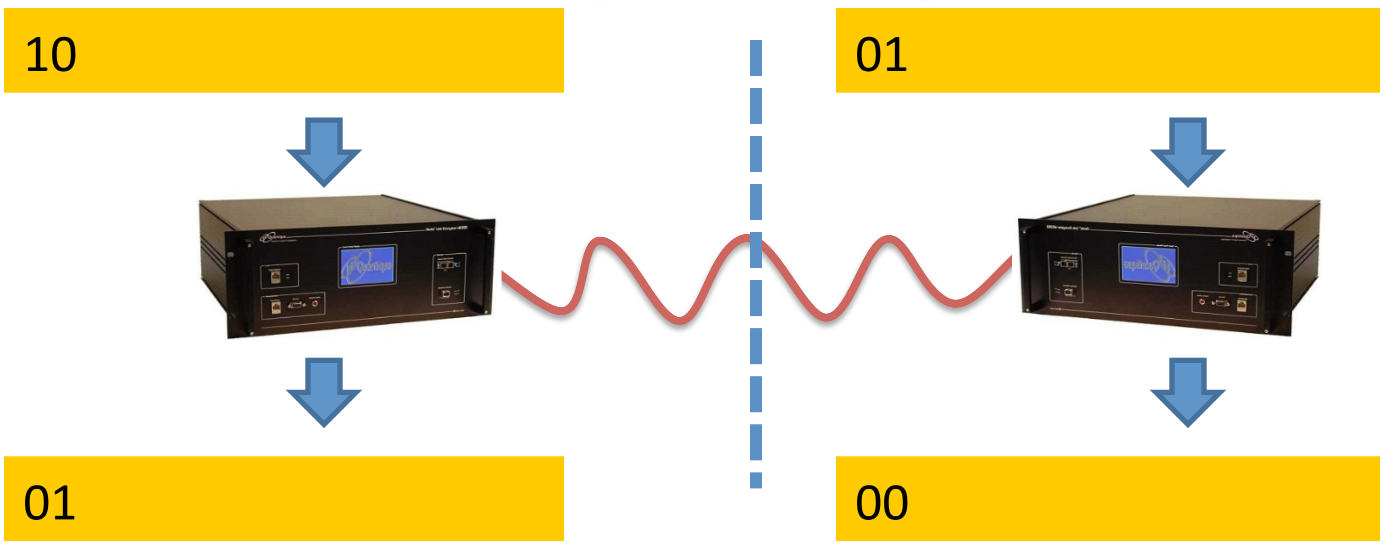
Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

Short random seed
m bits long

1000101001



n-bit inputs to devices are
deterministic function of seed S



Referee gives *n* bits to devices sequentially, and
collects *n* bits of output sequentially

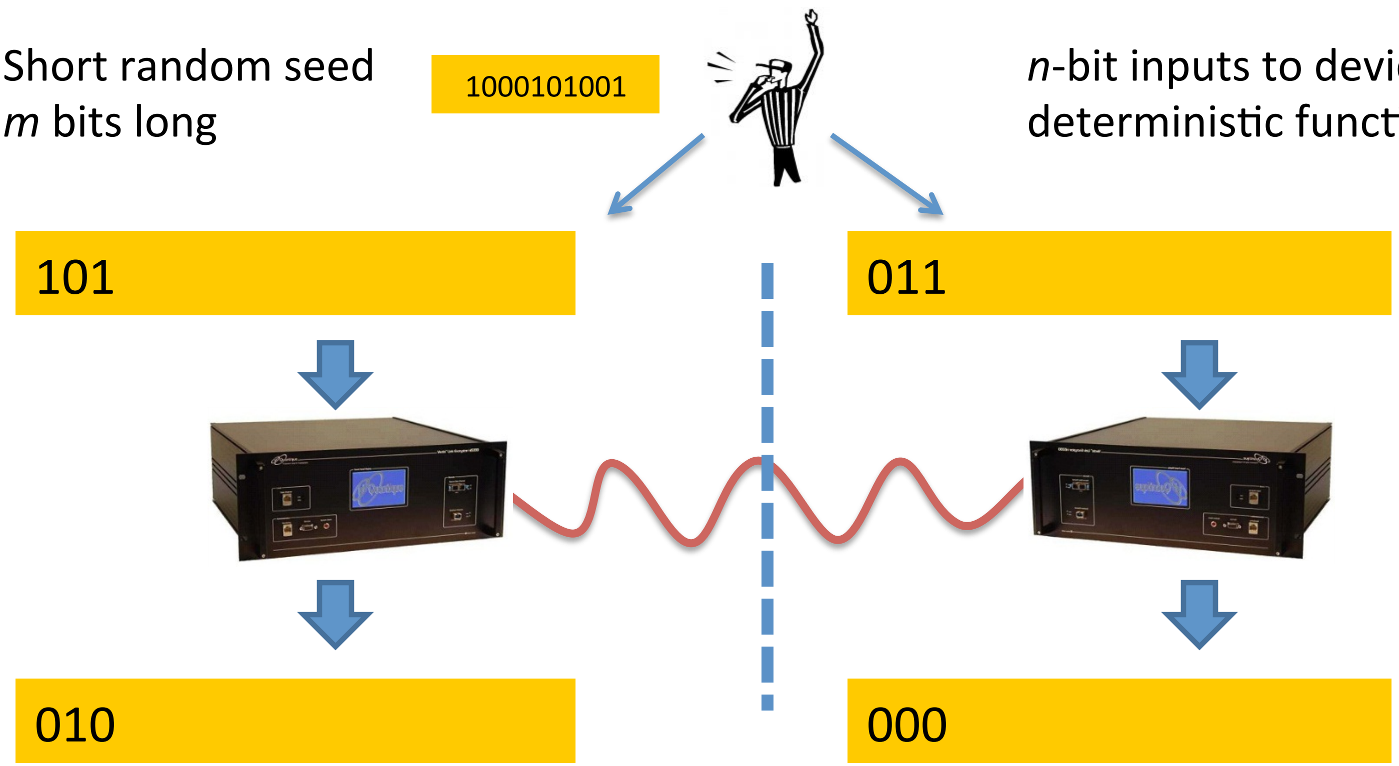
$n >> m$

Randomness expansion protocols

Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

Short random seed
m bits long

n-bit inputs to devices are
deterministic function of seed S



Referee gives *n* bits to devices sequentially, and
collects *n* bits of output sequentially

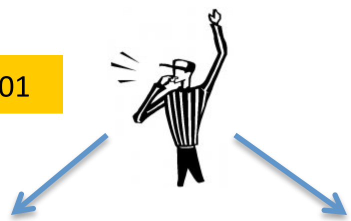
$n >> m$

Randomness expansion protocols

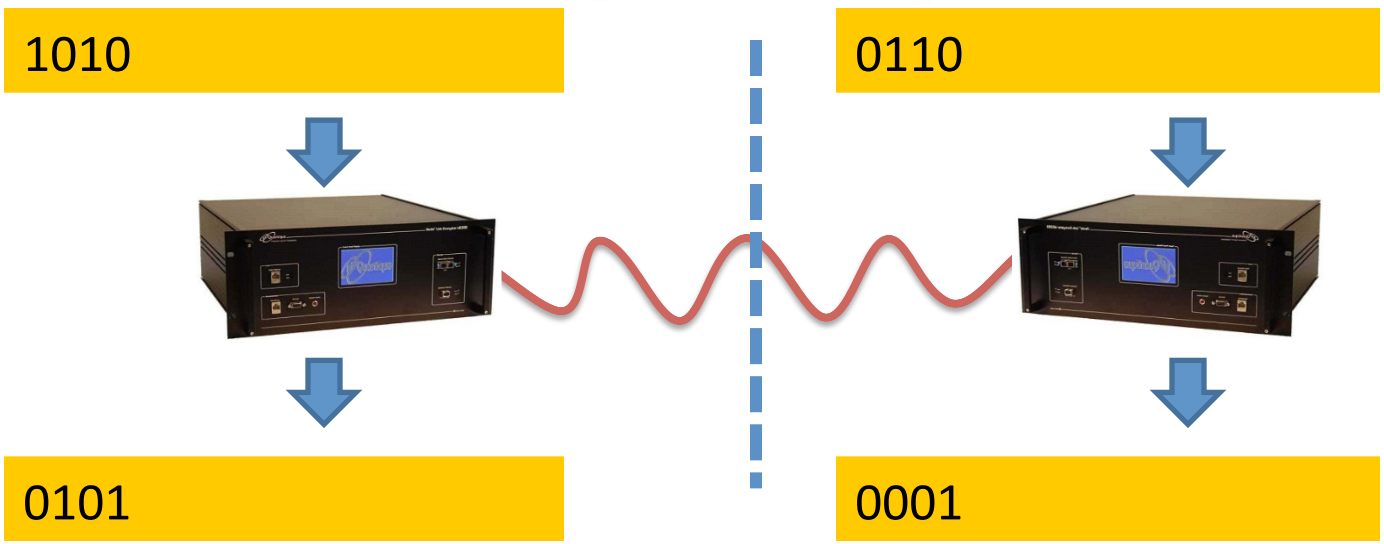
Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

Short random seed
m bits long

1000101001



n-bit inputs to devices are
deterministic function of seed S



Referee gives *n* bits to devices sequentially, and
collects *n* bits of output sequentially

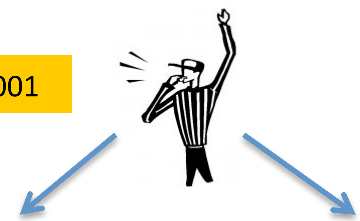
$n >> m$

Randomness expansion protocols

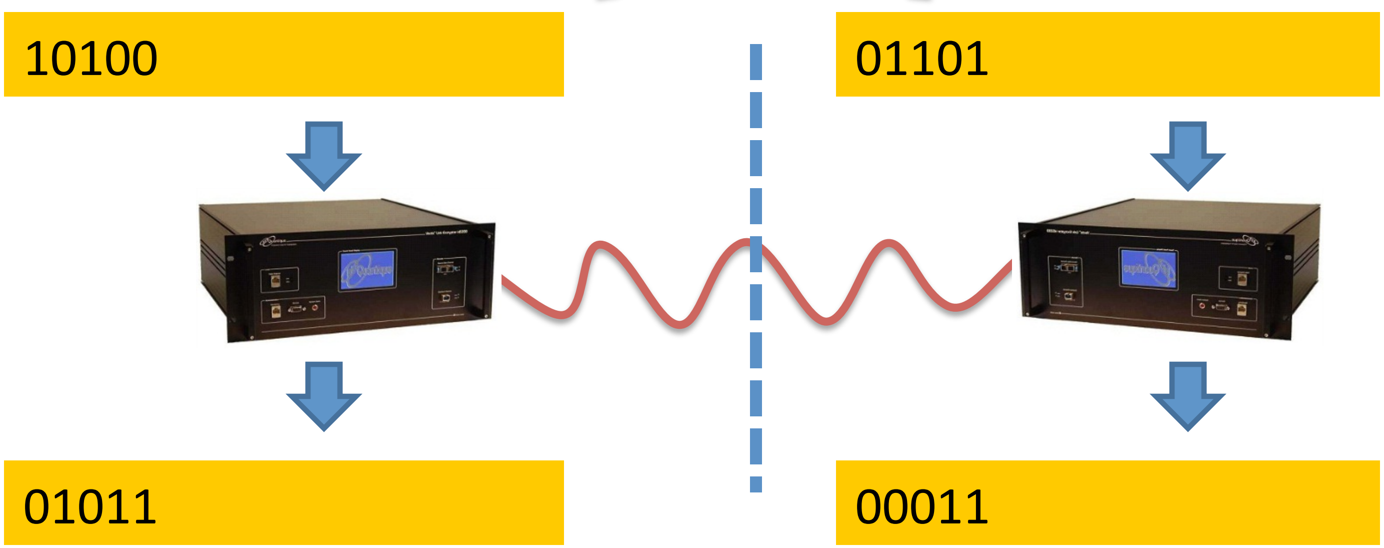
Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

Short random seed
m bits long

1000101001



n-bit inputs to devices are
deterministic function of seed S

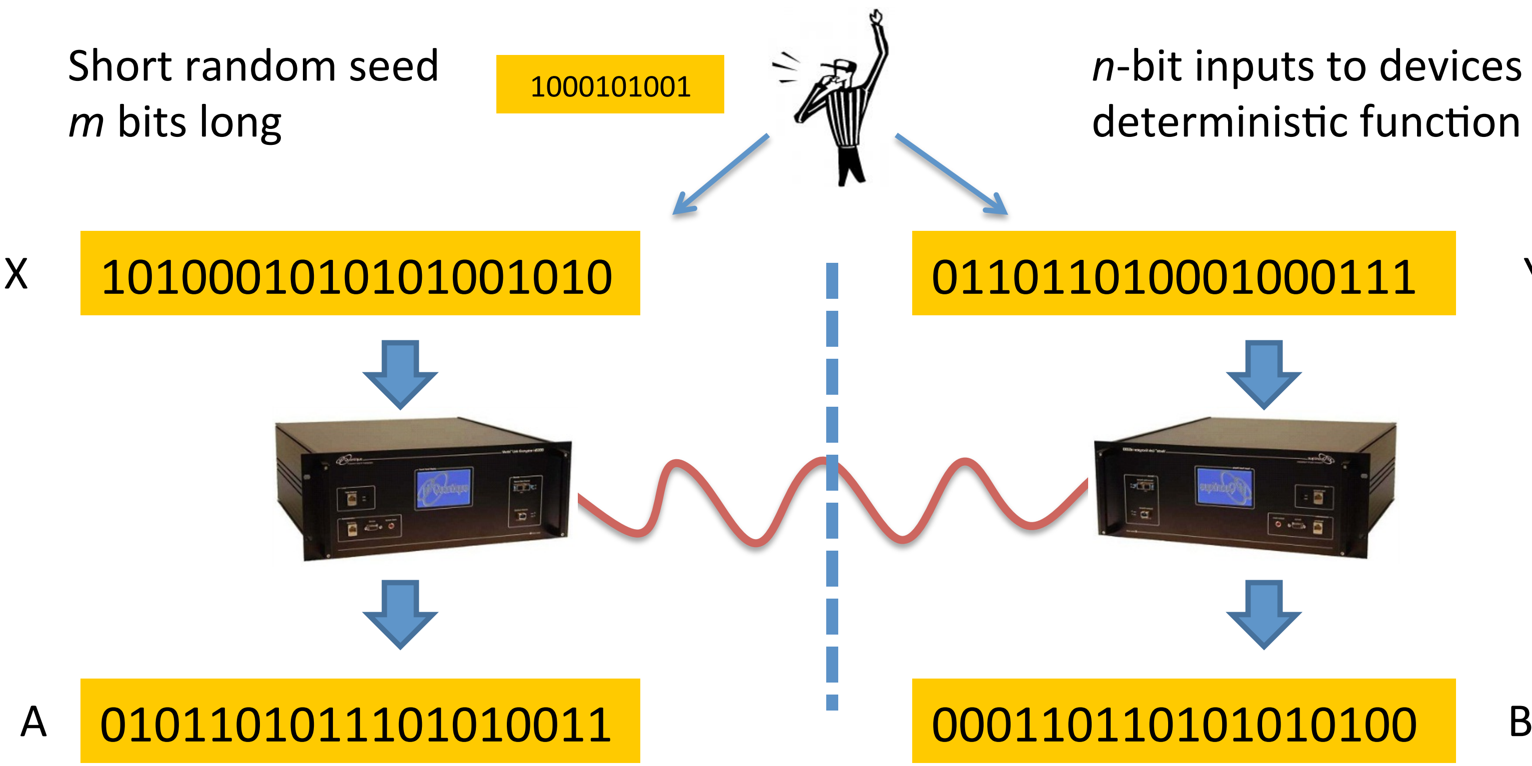


Referee gives *n* bits to devices sequentially, and
collects *n* bits of output sequentially

$n >> m$

Randomness expansion protocols

Model for protocols of [PAM+ '10][VV '12][CVY'13][MS'14]…



Referee tests inputs and outputs: T(X,Y,A,B) = 1?

e.g. T(X,Y,A,B) = 1 iff ~85% of $A_i + B_i = X_i Y_i$

Randomness expansion protocols

Model for protocols of [PAM+ ‘10][VV ’12][CVY’13][MS’14]…

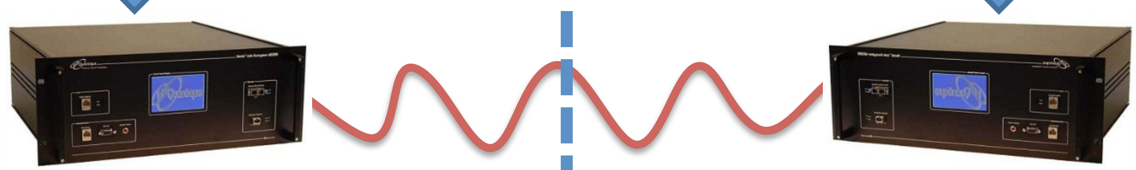
Short random seed
m bits long

1000101001

n-bit inputs to devices are
deterministic function of seed S

X 10100010101010001010

01101101000100001111 Y



A 01011010111010100011

00011011010101010100 B

Referee tests inputs and outputs: T(X,Y,A,B) = 1?
e.g. T(X,Y,A,B) = 1 iff ~85% of $A_i + B_i = X_i Y_i$

Outputs have Ω(*n*) bits of *certified* min-entropy!

An expanding list of randomness expansion protocols

- Roger Colbeck obtained *linear expansion (2006)*
  - $n = \theta(m)$
- Pironio, et al. achieved *quadratic expansion (2010)*
  - $n = \theta(m^2)$
- Vazirani-Vidick was first to achieve *(quantum-secure) exponential expansion (2012)*
  - $n = 2^{\Omega(m)}$

Is there a limit?

An expanding list of randomness expansion protocols

- Roger Colbeck obtained *linear expansion (2006)*
  - $n = \theta(m)$
- Pironio, et al. achieved *quadratic expansion (2010)*
  - $n = \theta(m^2)$
- Vazirani-Vidick was first to achieve *(quantum-secure) exponential expansion (2012)*
  - $n = 2^{\Omega(m)}$

Is there a limit?

[CVY'13]: for a broad class of **non-adaptive** protocols, **exp(exp(m))** expansion is the limit! This is due to **cheating strategies.**

An expanding list of randomness expansion protocols

- Roger Colbeck obtained *linear expansion (2006)*
  - $n = \theta(m)$
- Pironio, et al. achieved *quadratic expansion (2010)*

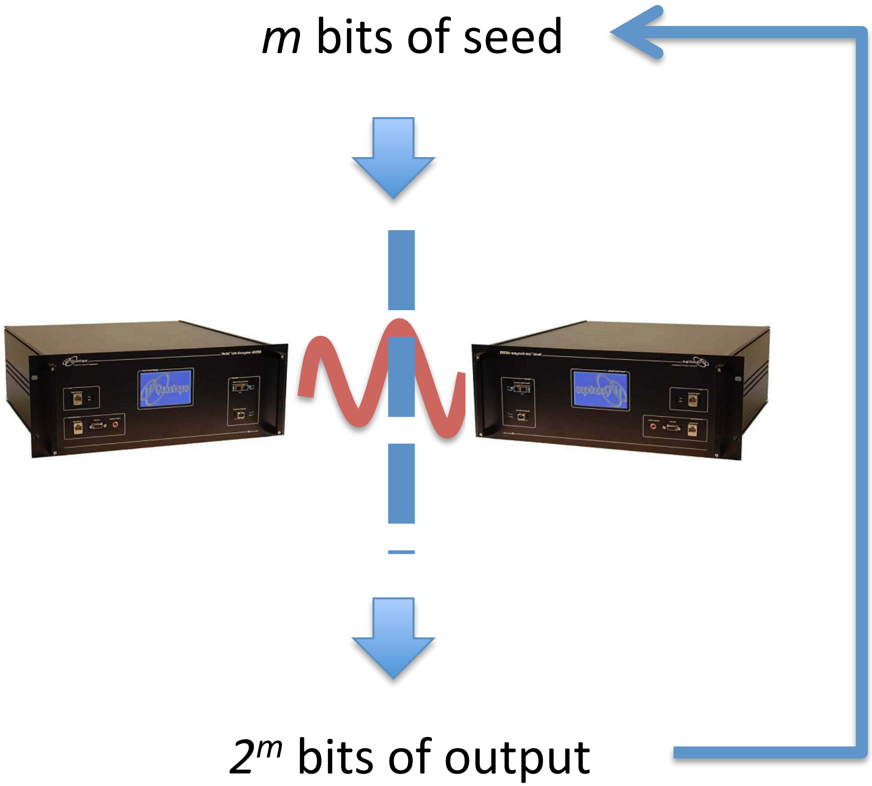
Okay, what about adaptive protocols?

- Vazirani-Vidick was first to achieve *(quantum-secure) exponential expansion (2012)*
  - $n = 2^{\Omega(m)}$

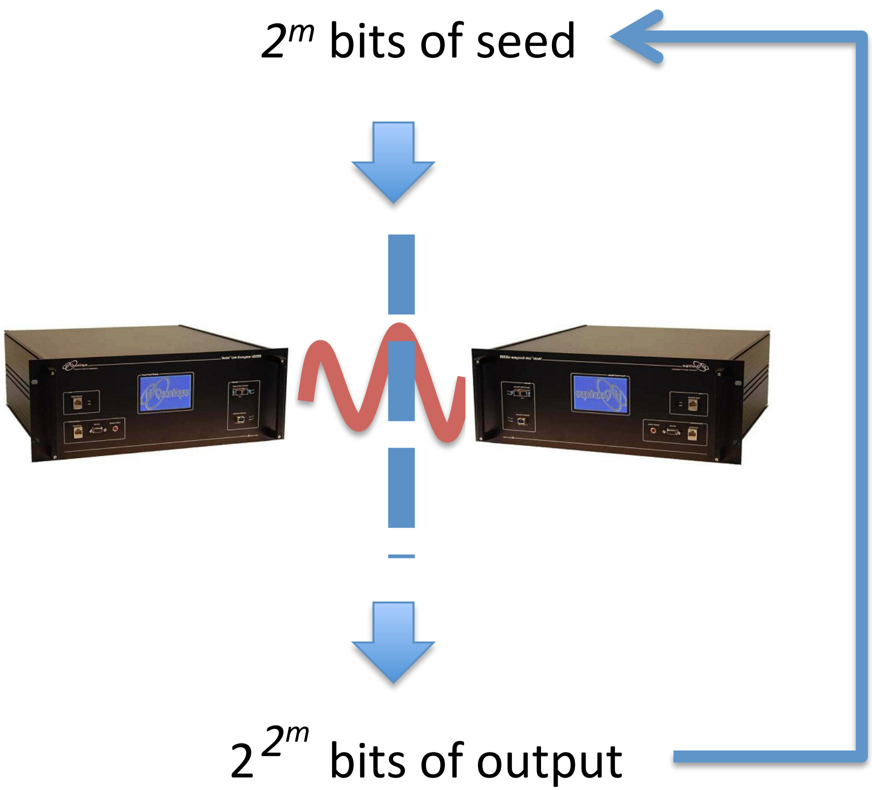
Is there a limit?

[CVY'13]: for a broad class of **non-adaptive** protocols, **exp(exp(m))** expansion is the limit! This is due to **cheating strategies.**

First attempt

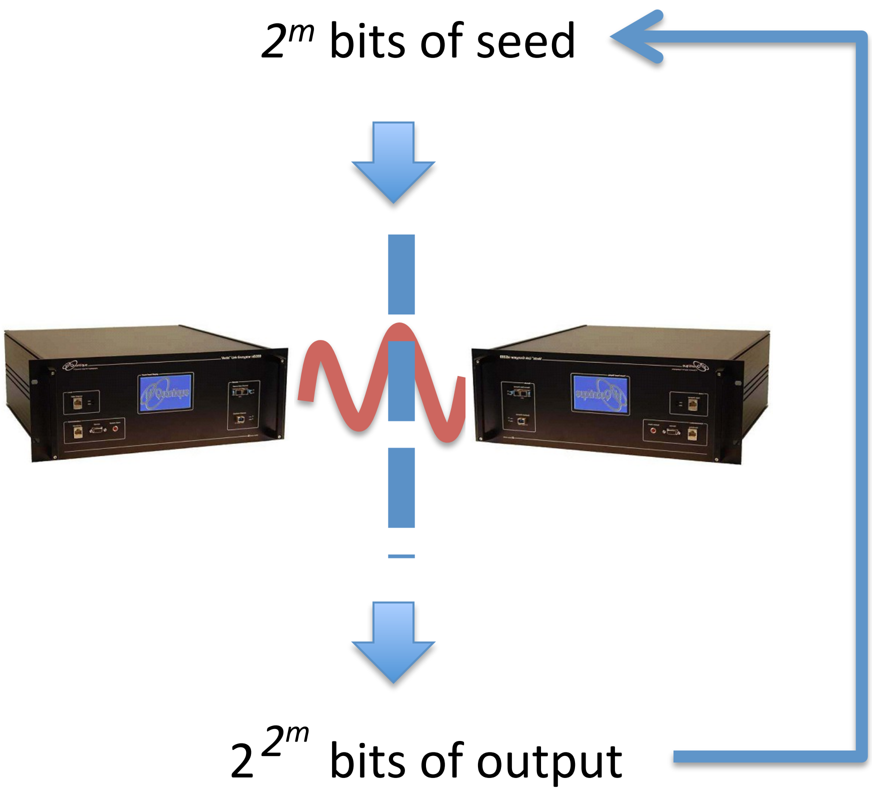


First attempt



And so on….

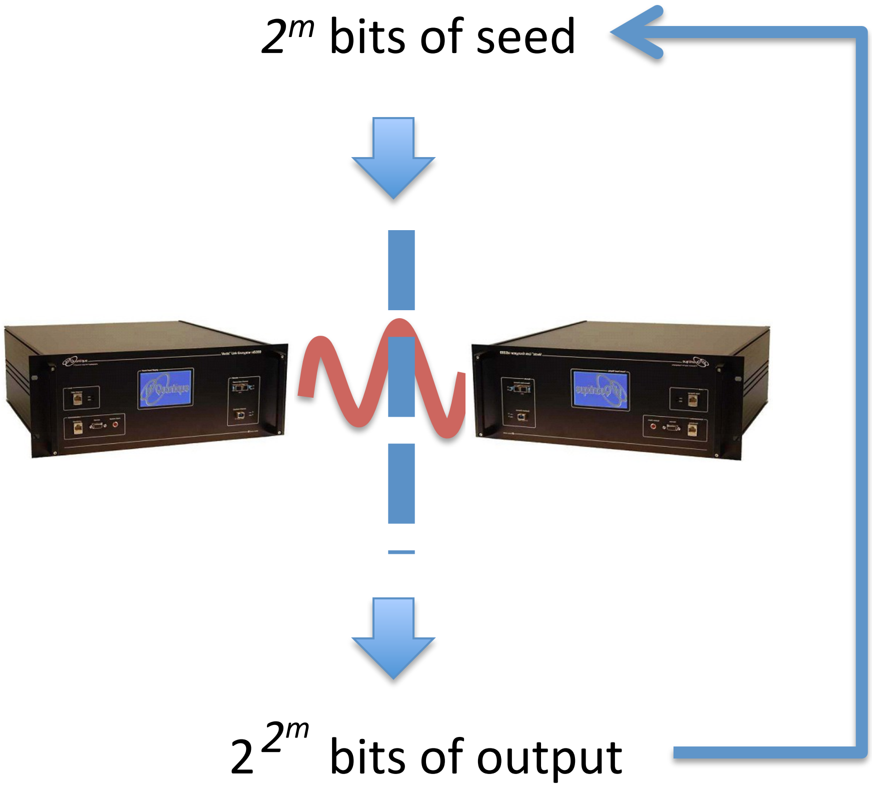
First attempt



The outputs are **not** uniform and independent of the devices: devices may take be able to predict future inputs!

And so on….

First attempt



The outputs are **not** uniform and independent of the devices: devices may take be able to predict future inputs!

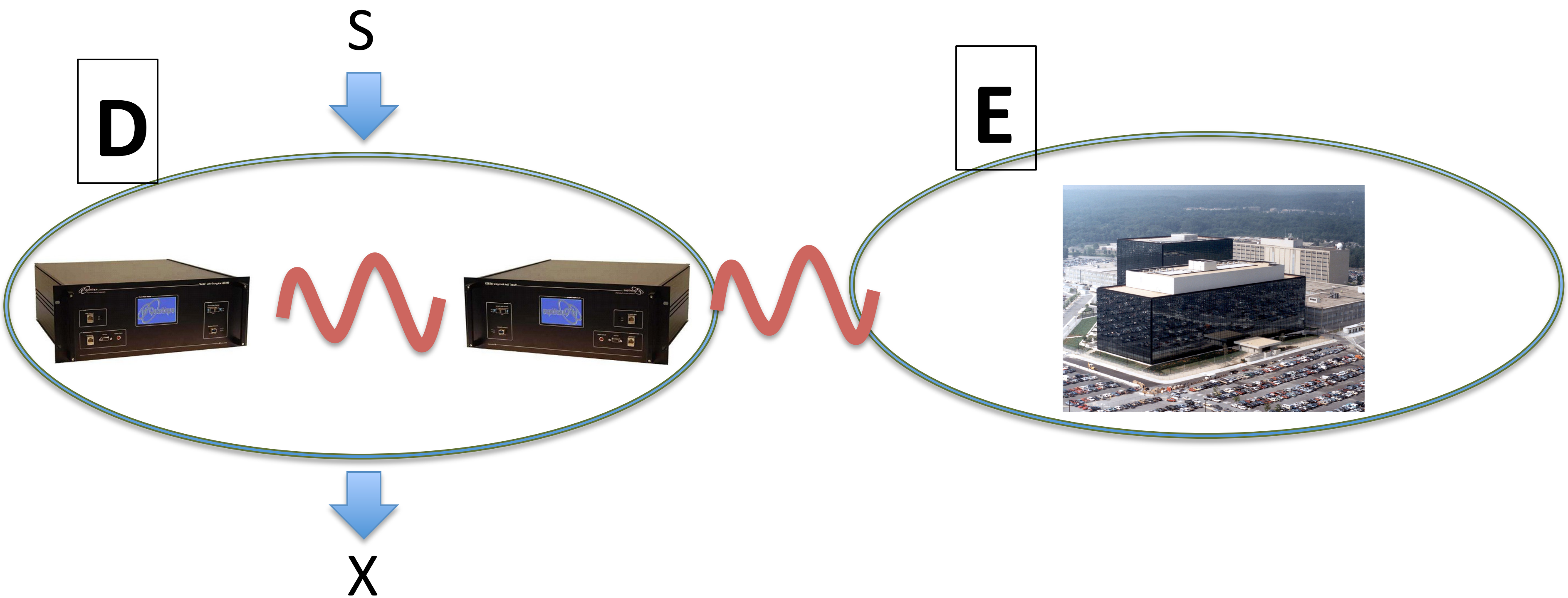
What about variants, such as XORing together Alice and Bob’s outputs? Or applying more complicated post-processing?

I don’t know how to analyze this…

And so on….

Second attempt

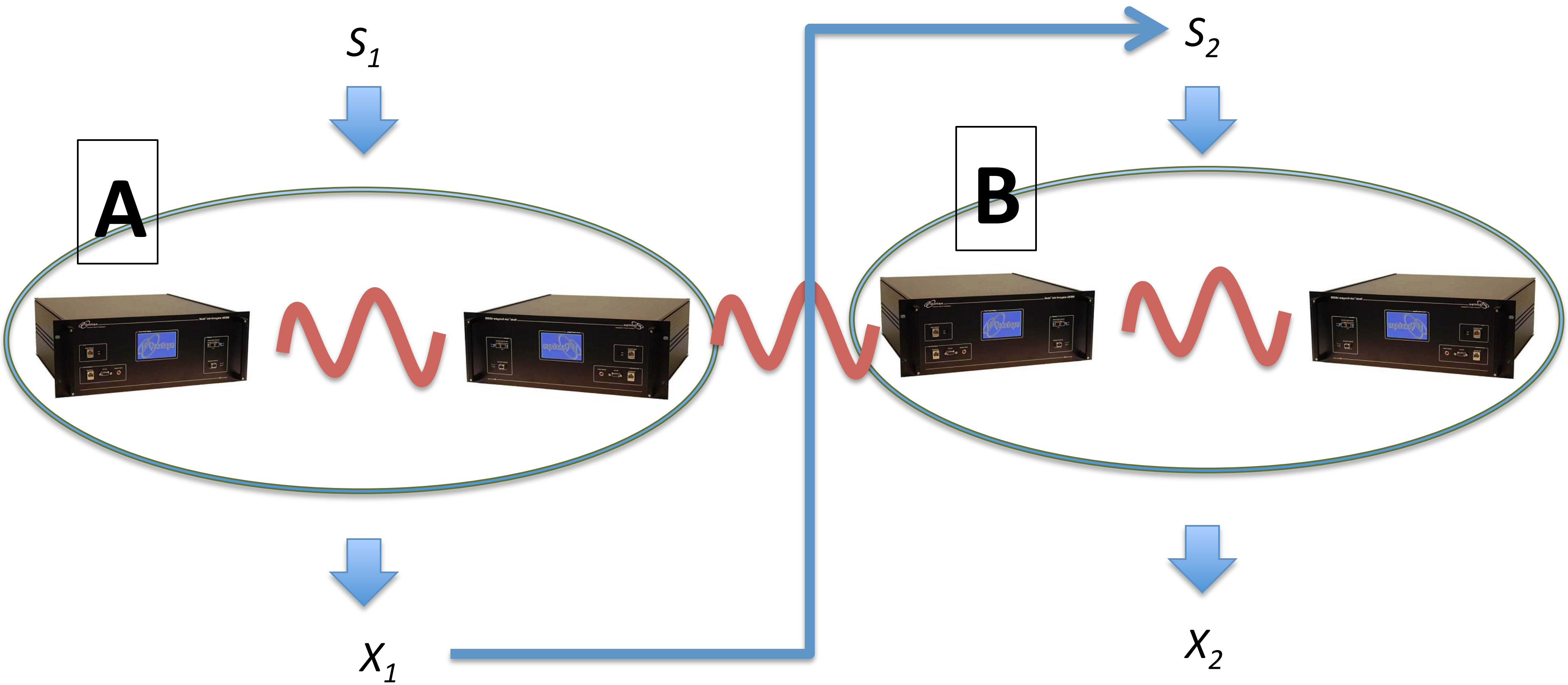
Use the fact that the [VV12] protocol is **quantum-secure**:



$$\rho_{SDE} = U_m \otimes \rho_{DE} \Rightarrow \rho_{XE} \approx U_n \otimes \rho_E$$

Second attempt

$$\rho_{S_1 AB} = U_m \otimes \rho_{AB}$$

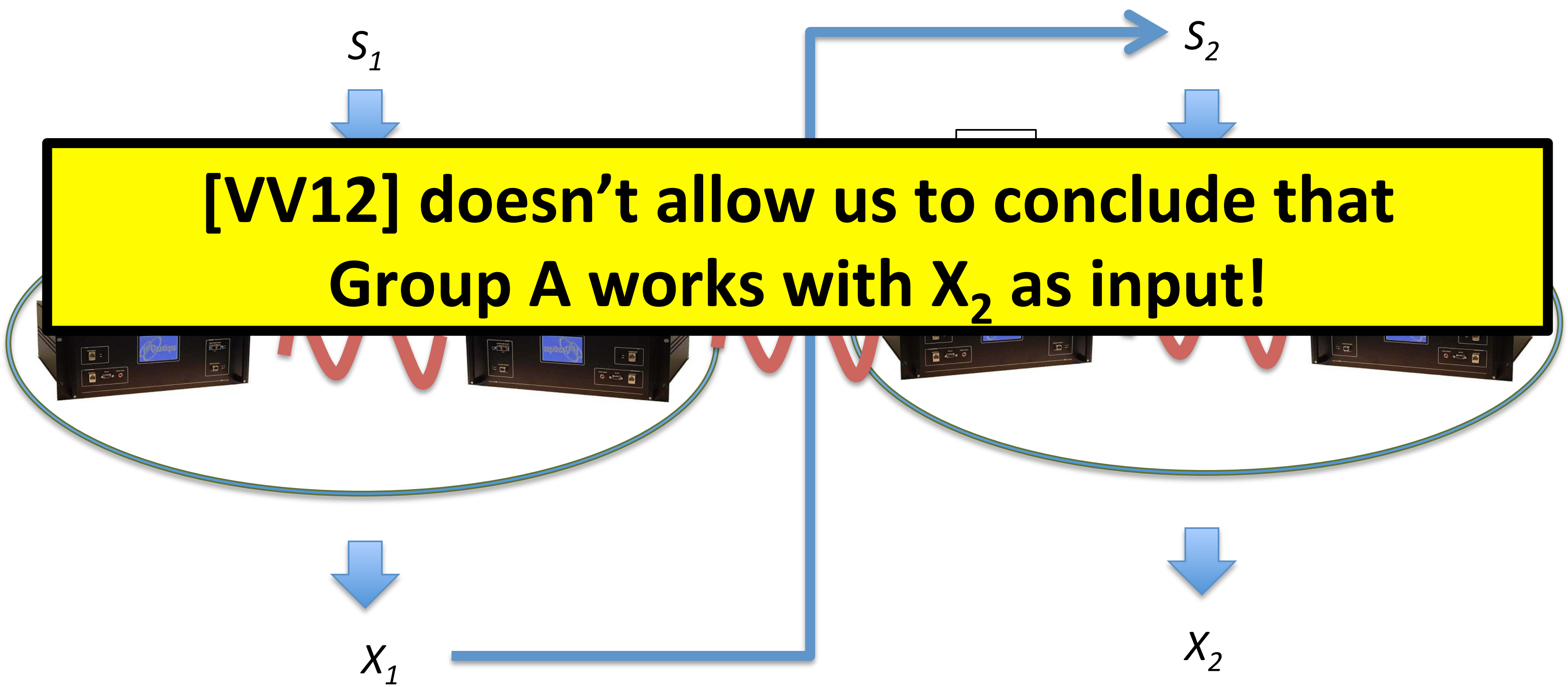


$$\rho_{X_1 B} \approx U_{2^m} \otimes \rho_B$$

$$\rho_{X_2} \approx U_{2^{2^m}}$$

Second attempt

$$\rho_{S_1 AB} = U_m \otimes \rho_{AB}$$

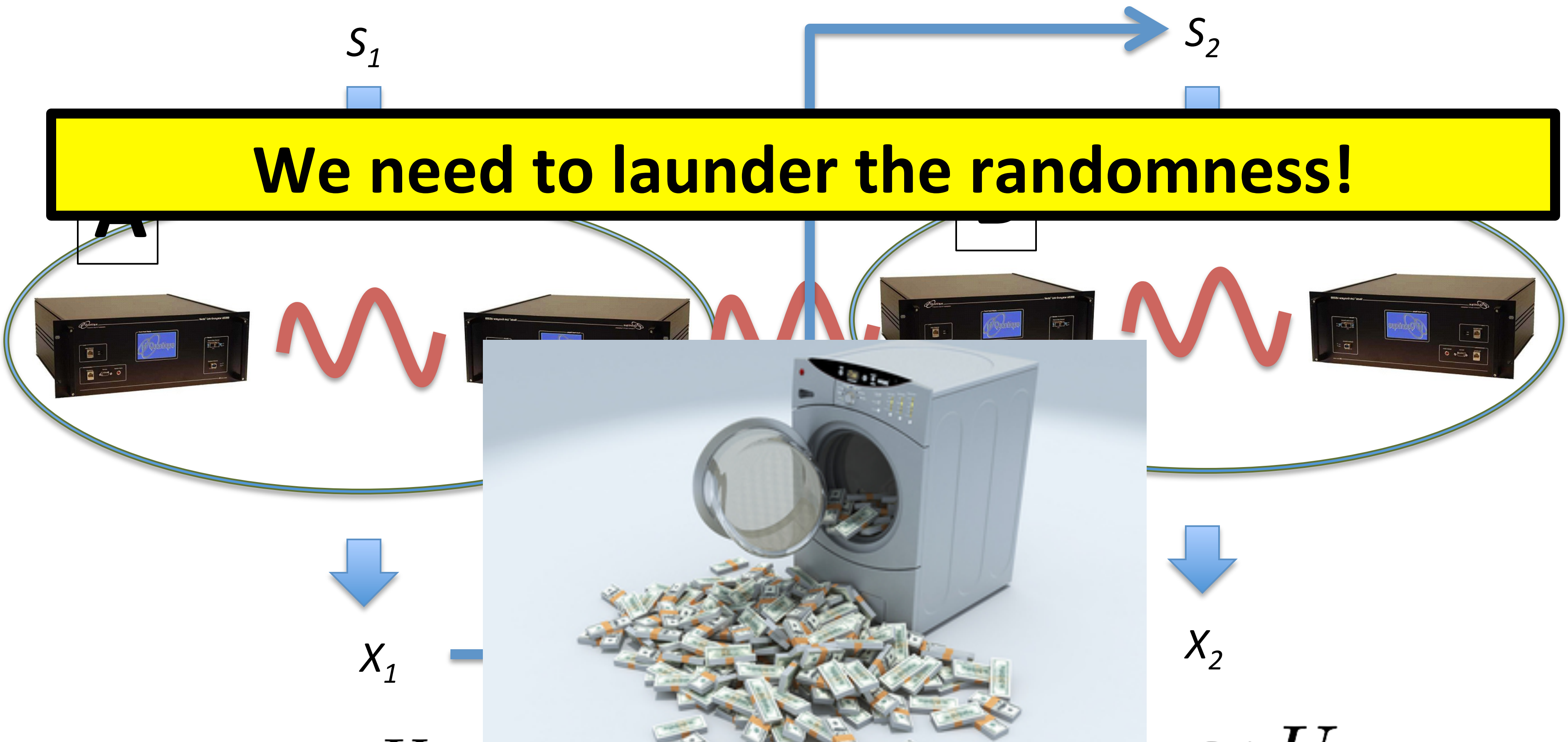


[VV12] doesn't allow us to conclude that Group A works with X_2 as input!

$$\rho_{X_1 B} \approx U_{2^m} \otimes \rho_B \qquad \rho_{X_2} \approx U_{2^{2^m}}$$

Second attempt

$$\rho_{S_1AB} = U_m \otimes \rho_{AB}$$



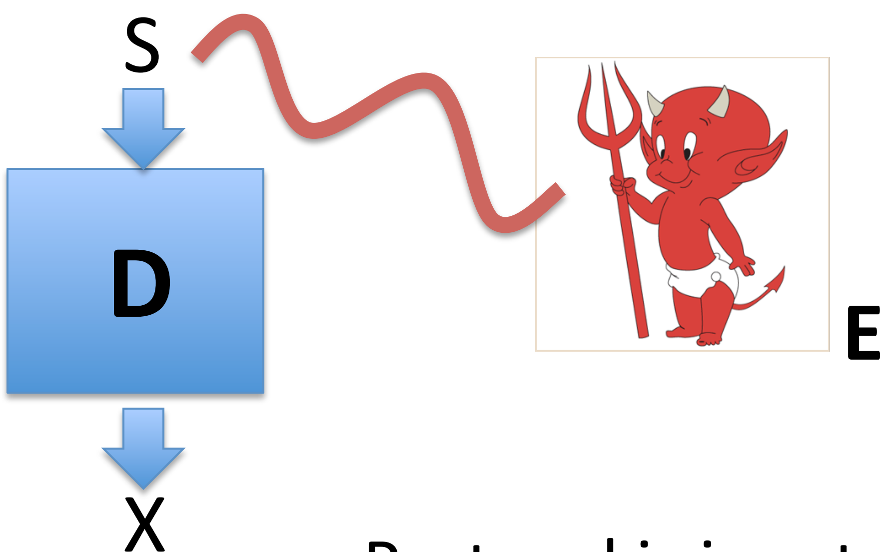
We need to launder the randomness!

$$\rho_{X_1B} \approx U_{2^m} \otimes \rho_B$$

$$\rho_{X_2} \approx U_{2^{2m}}$$

Input Security

Input Secure Protocol: input to protocol can be correlated with eavesdropper, but output is not!



Protocol is input secure if:

$$\rho_{SD} = U_m \otimes \rho_D \Rightarrow \rho_{XE} \approx U_n \otimes \rho_E$$

Are there Input Secure protocols?

- Until recently, this was not clear.
- Note: extractors are *not* Input Secure.

Quantum-Secure Extractor: $\mathrm{Ext} : \{0,1\}^m \times \{0,1\}^d \to \{0,1\}^n$

$$\rho_{SDE} = U_d \otimes \rho_{DE} \qquad H_{\min}(D|E) \geq k$$

⇓

$$\rho_{\mathrm{Ext}(D,S)SE} \approx U_n \otimes \rho_S \otimes \rho_E$$

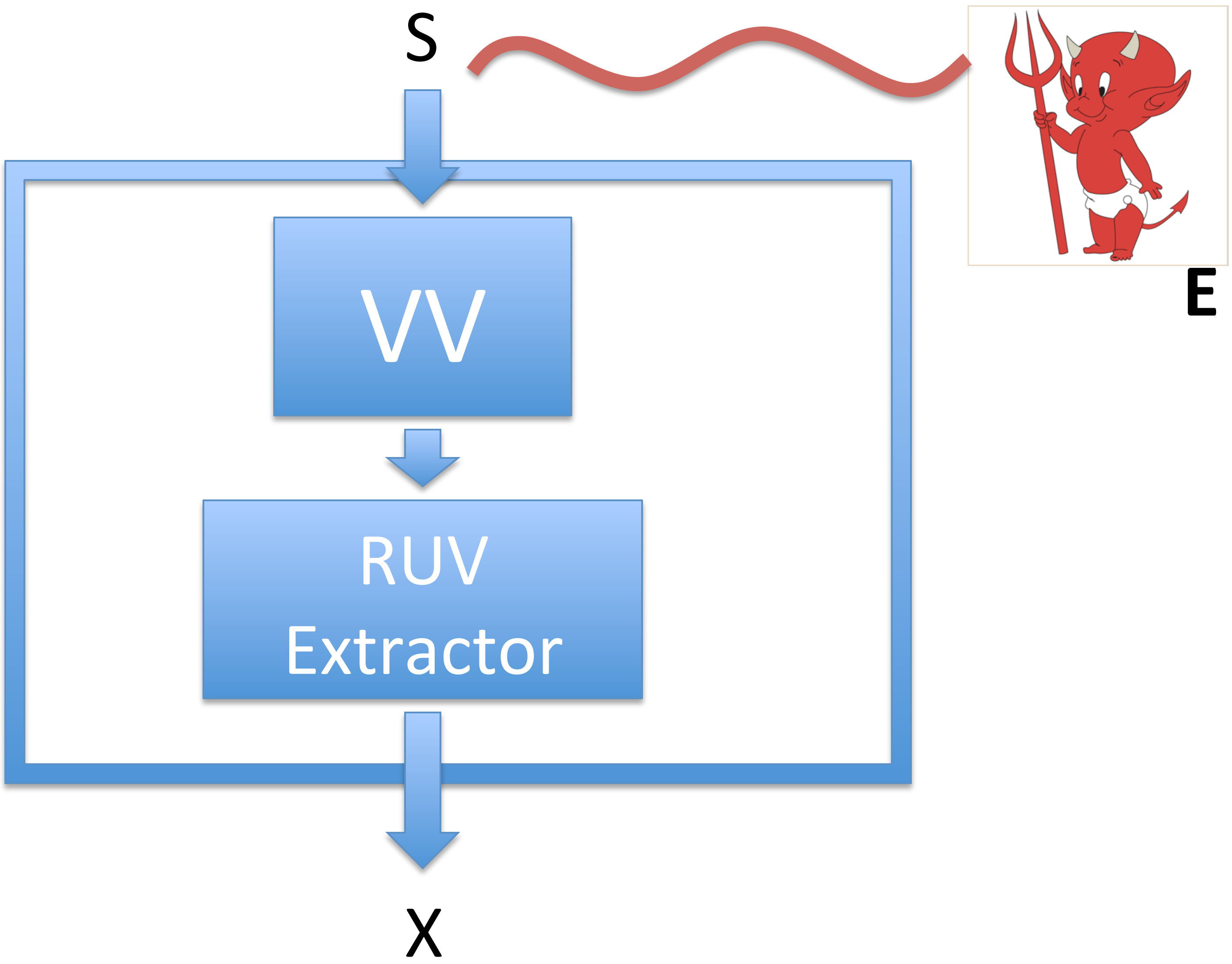
- Used at the end of randomness expansion protocols to create near-uniform, private randomness (provided extractor seed is not known to the adversary)
- Counter-example:

$$E = (S, \mathrm{Ext}(D,S)_1)$$
$$H_{\min}(D|E) \geq n - O(\log n)$$

but

$$\rho_{\mathrm{Ext}(D,S)E} \not\approx U_n \otimes \rho_E$$

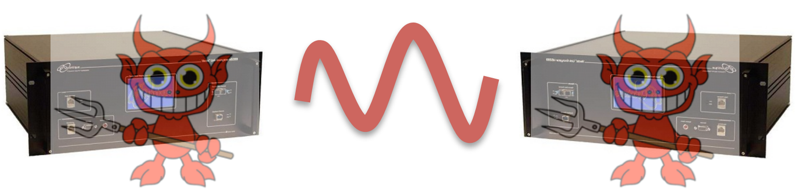
Our Input Secure protocol



Rigidity of CHSH games

CHSH Rigidity [Mayers, Yao ‘03][MKS12][YN13]

If two isolated devices win the CHSH game with ~85% probability, then they must be using a strategy that is very close to the *ideal, canonical* CHSH strategy.



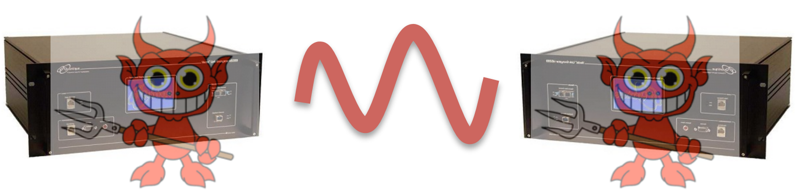
$$|\psi\rangle$$

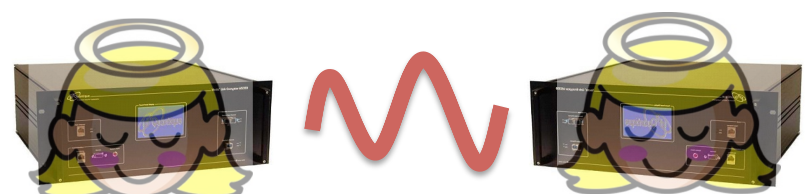
Devices win ~85% of the time!

Rigidity of CHSH games

CHSH Rigidity [Mayers, Yao ‘03][MKS12][YN13]

If two isolated devices win the CHSH game with ~85% probability, then they must be using a strategy that is very close to the *ideal, canonical* CHSH strategy.





$$|\psi\rangle \cong \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \otimes |\text{junk}\rangle$$

Devices win ~85% of the time!

Multigame CHSH Rigidity

x	y	a	b
0	1	0	0
1	1	0	1
0	0	1	1
1	1	1	0
0	1	1	1
1	1	0	1
0	0	0	1
1	1	0	1

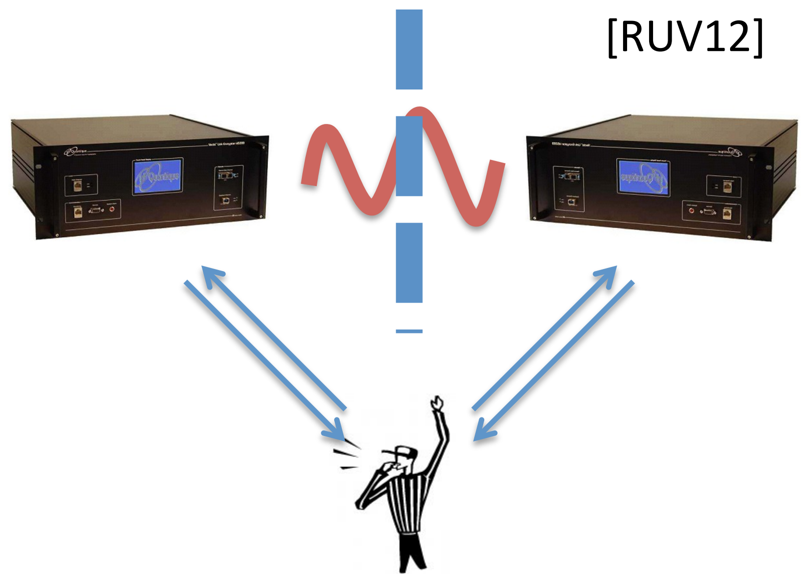
N

} Random block of games

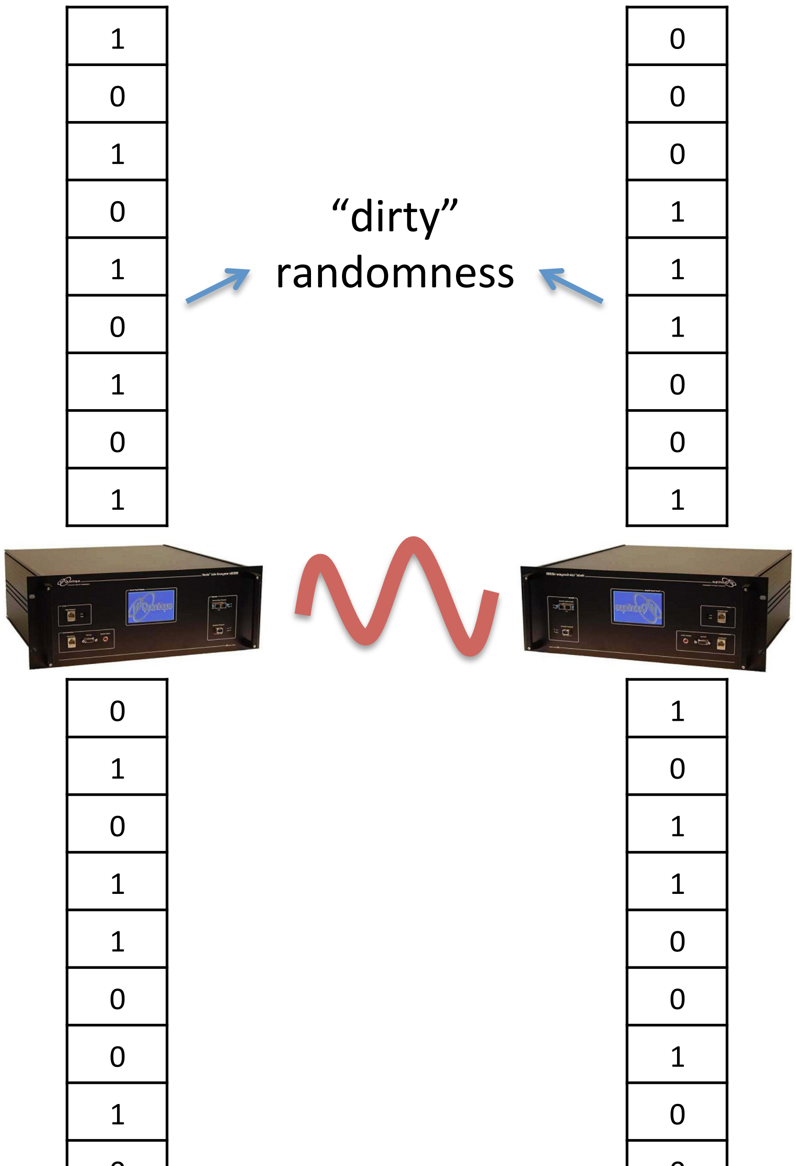
Multigame CHSH Rigidity

If two isolated devices play *N* **sequential** CHSH games, and consistently win ~85% of the games, then w.h.p. a random block of games (N^c for some $0 < c < 1$) were played using a strategy approx. isomorphic to the ideal product strategy!

[RUV12]

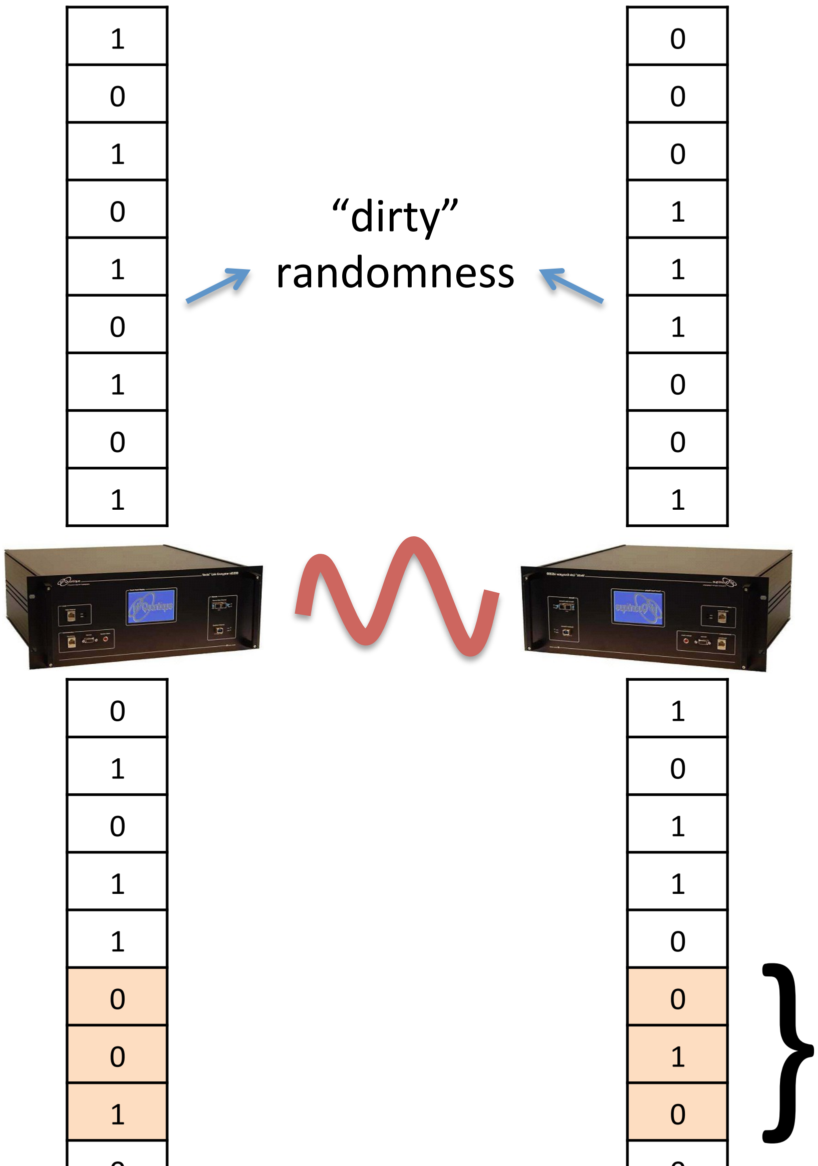


How to launder randomness



Win ~85% of games? ✓

How to launder randomness



Win ~85% of games? ✓

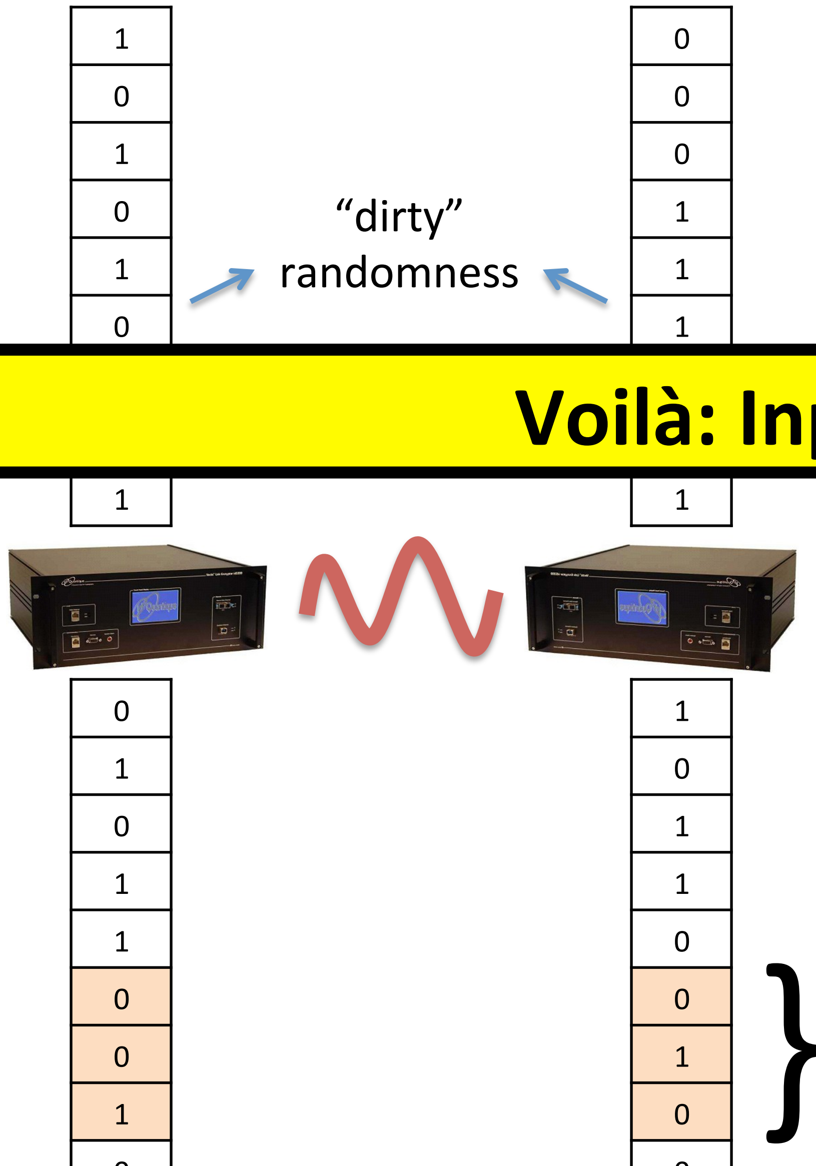
Select a random block of games

The block of bits are (approx.)
- Uniformly random
- Unentangled/uncorrelated with any eavesdropper

↑

W.h.p., block of games was played using (approx.) the ideal CHSH strategy.

How to launder randomness



Win ~85% of games? ✓

Select a random block of games

Voilà: Input Security!

The block of bits are (approx.)

- Uniformly random
- Unentangled/uncorrelated with any eavesdropper

W.h.p., block of games was played using (approx.) the ideal CHSH strategy.

Not so fast...

- Technical concerns

  1. Conditioned on passing the RUV protocol, an ideal block may not be secure!

Worry: Conditioning on passing the protocol can introduce correlations, despite the use of an ideal strategy.

Example: Alice and Bob could use ideal strategy in Blocks 1, 2, and 3.

If XOR of Alice's output in Block 1 is 0, then Alice fails all games after Block 4.

Otherwise, Alice plays honestly.

Conditioned on passing ~85% of games, Alice's output in Block 1 is far from uniform!

Not so fast…

- Technical concerns

  1. Conditioned on passing the RUV protocol, an ideal block may not be secure!

Worry: Conditioning on passing the protocol can introduce correlations, despite the use of an ideal strategy.

Resolution: If Pr(Pass RUV) is not too small, then conditioning cannot skew the distribution of too many blocks.

Before conditioning: $I(X:E) \approx 0$

$$\Rightarrow I(X:EF) \lesssim 2H(F) \leq 2$$

Chain rule: $I(X:EF) = \sum_i I(X_i : EF | X_{<i})$

$$\geq \sum_i I(X_i : EF)$$

Most blocks are unaffected by conditioning!

$$\Rightarrow \mathbb{E}[I(X_i : EF)] \lesssim 2/B$$

Not so fast…

- Technical concerns
  1. Conditioned on passing the RUV protocol, an ideal block may not be secure!
  2. Who chooses the random blocks?

Worry: Adversary can select non-ideal blocks, or other bad blocks.

N

x	y	a	b
0	1	0	0
1	1	0	1
0	0	1	1
1	1	1	0
0	1	1	1
1	1	0	1
0	0	0	1
1	1	0	1

"Random" block of games

E

Not so fast…

- Technical concerns
  1. Conditioned on passing the RUV protocol, an ideal block may not be secure!
  2. Who chooses the random blocks?

N

x	y	a	b
0	1	0	0
1	1	0	1
0	0	1	1
1	1	1	0
0	1	1	1
1	1	0	1
0	0	0	1
1	1	0	1

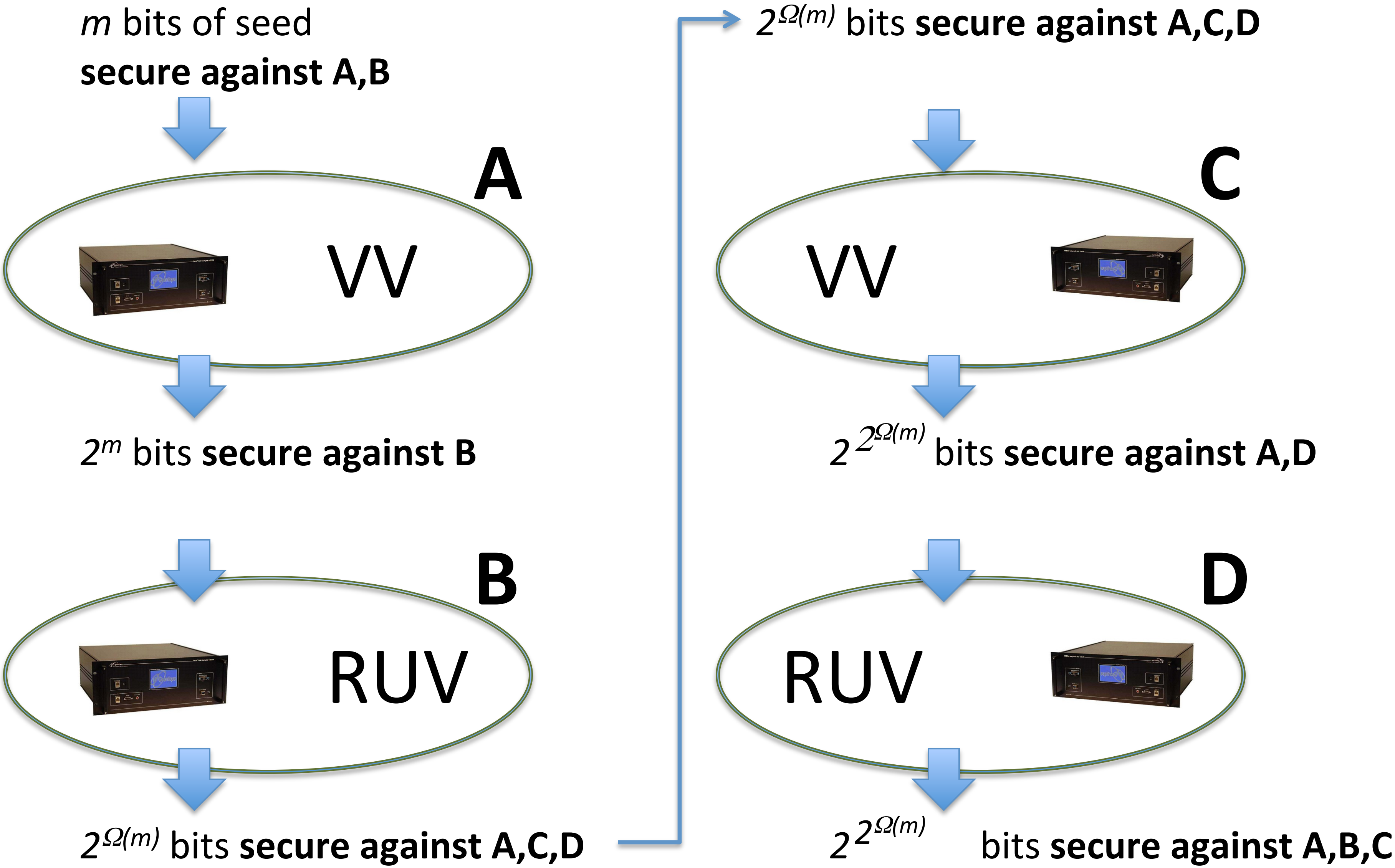
"Random" block of games

Worry: Adversary can select non-ideal blocks, or other bad blocks.

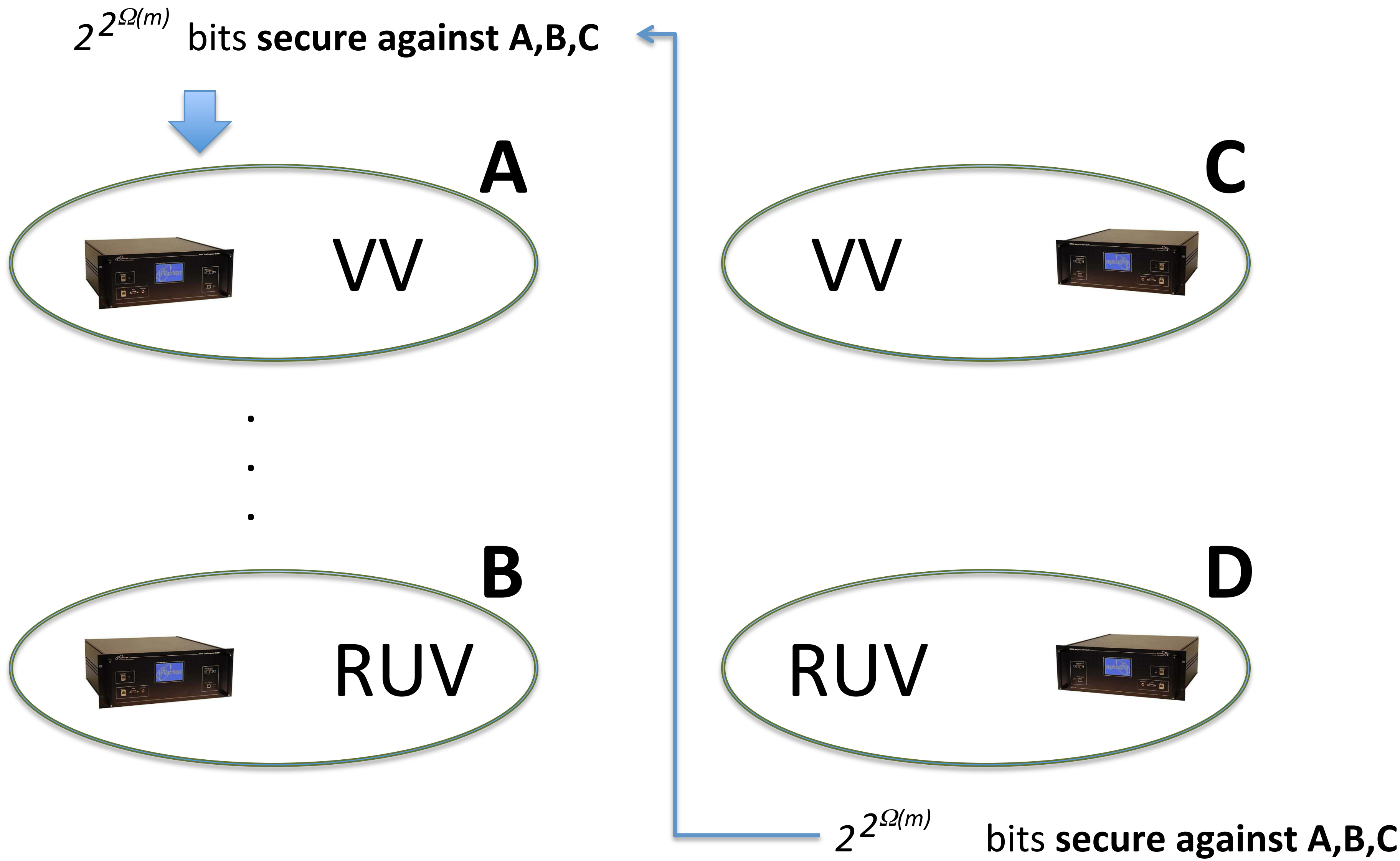
E

Resolution: Can't happen using a local simulation argument.

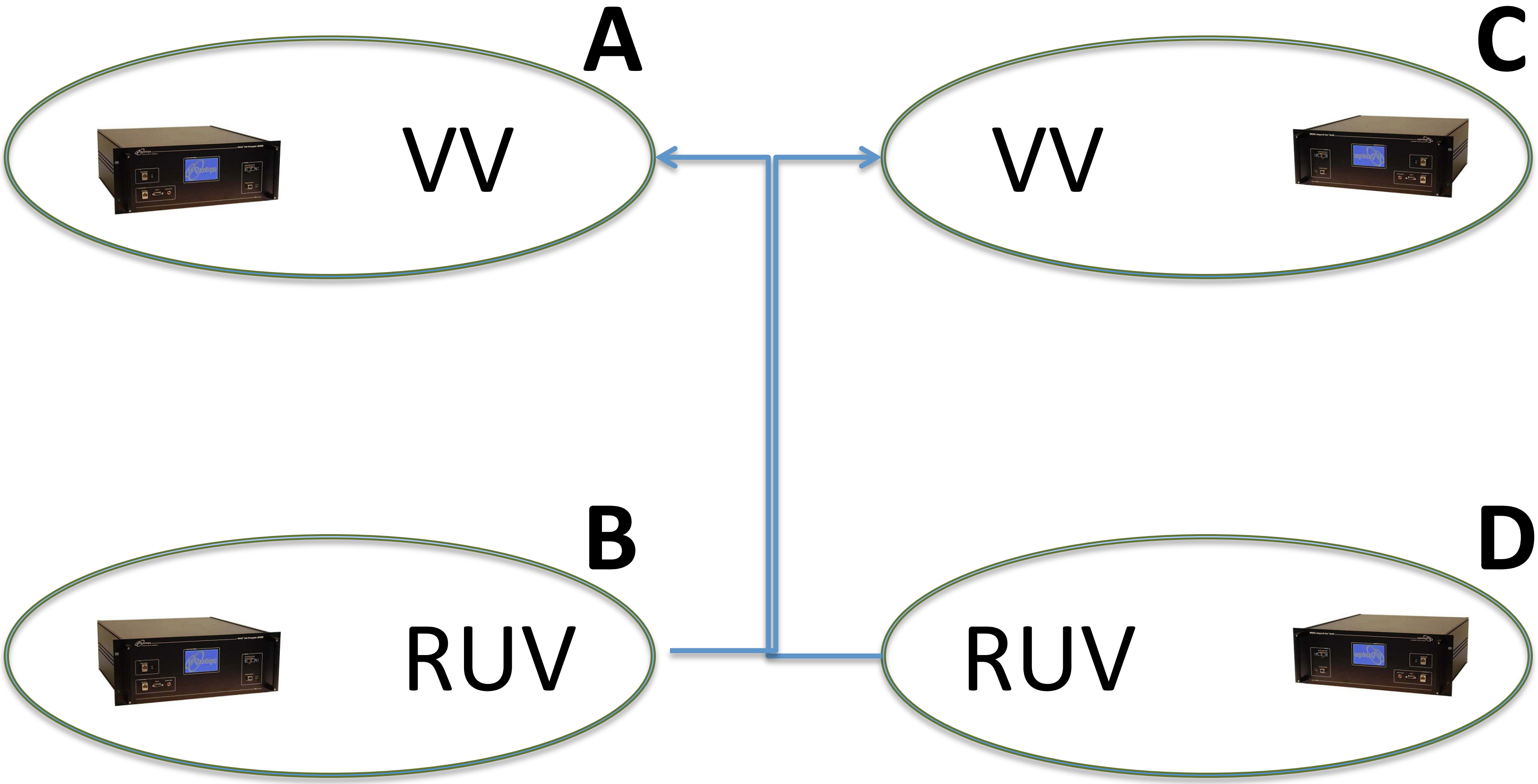
Final protocol



Final protocol



Final protocol



Equivalence Lemma

[Chung, Shi, Wu ’14]

Expansion protocol requiring “globally secure” input:

$$\rho_{SDE} = U_m \otimes \rho_{DE} \Rightarrow \rho_{XSE} \approx U_n \otimes \rho_{SE}$$

…does not require input to be secure against eavesdropper (i.e. Input Secure)

$$\rho_{SD} = U_m \otimes \rho_{D} \Rightarrow \rho_{XSE} \approx U_n \otimes \rho_{SE}$$

So [VV’12] and [MS’14] protocols are also Input Secure!

Note: cannot be applied to randomness extractors!

Open Questions

For "Science advocates"

- Robust randomness expansion?
  - [CVY'13] [MS'14] made progress in this direction
- Quantum-secure randomness expansion with inefficient detectors
- What if we allow devices to leak *k* bits during protocol?
- Applications/Generalizations of Input Security?

For "Scientists"

- Infinite expansion with 2 devices?