Universal Reductions

Reductions relative to stateful oracles

Benjamin Chan

Cornell Tech

November 10 2022

Joint work with Cody Freitag & Rafael Pass

Suppose we had a weak OWF **f**

f

$\forall$ A,
Pr[A inverts f] < 3/4

We want to build a strong OWF **f'**

f'

$\forall$ A,
Pr[A inverts f] = negl(.)

How do we prove the security of **f’**?

How do we prove the security of **f'**? By security reduction.

Suppose ∃ some **A** that inverts **f'** with 1/poly probability:

$$\mathbf{f'} \rightleftarrows \mathbf{A}$$

How do we prove the security of **f'**? By security reduction.

Suppose ∃ some **A** that inverts **f'** with 1/poly probability:

$$\mathbf{f'} \rightleftarrows \mathbf{A}$$

Then ∃ **A'** that inverts **f** with probability >3/4:

$$\mathbf{f} \rightleftarrows \mathbf{A'}$$

How do we prove the security of f'? By security reduction.

Suppose ∃ some **A** that inverts **f'** with 1/poly probability:

$$f' \rightleftarrows A$$

Then ∃ **A'** that inverts **f** with probability >3/4:

$$f \rightleftarrows A'$$

Contradiction! (with the weak one-wayness of f)

How do we prove the security of f’? By security reduction.

Suppose ∃ some **A** that inverts **f’** with 1/poly probability:

$$\mathbf{f'} \rightleftarrows \mathbf{A}$$

Then ∃ **A’** that inverts **f** with probability >3/4:

$$\mathbf{f} \rightleftarrows \mathbf{A'}$$

Contradiction! (with the weak one-wayness of f)

Observe:
This proof is only useful “in the real world” if our model for attackers correctly captures the behavior of “real-life” adversaries!

How do we prove the security of f'? By security reduction.

Suppose ∃ some **A** that inverts **f'** with 1/poly probability:

$$f' \rightleftarrows A$$

Then ∃ **A'** that inverts **f** with probability >3/4:

$$f \rightleftarrows A'$$

Contradiction! (with the weak one-wayness of f)

Extended Physical Church Turing Hypothesis:
All "real-life" attackers are captured by PPT (resp. QPT) Turing Machines

How do we prove the security of f’? By security reduction.

Suppose ∃ some PPT **A** that inverts f’ with 1/poly probability:

$$f' \rightleftarrows A$$

Then $\mathbf{R}^{A}$ inverts **f** with probability >3/4:

$$f \rightleftarrows R^{\boxed{A}}$$

Classically, we can write black-box reductions R^A:
R queries A many times

Contradiction! (with the weak one-wayness of f)

How do we prove the security of **f'**? By security reduction.

Suppose ∃ some PPT **A** that inverts f' with 1/poly probability:

$$f' \rightleftarrows A$$

Then **R**^{A} inverts **f** with probability >3/4:

$$f \rightleftarrows R^{\boxed{A}}$$

Classically, we can write black-box reductions R^A:
R queries A many times

say we want to invert **y** = f(x):

A(~, ~, **y**, ~, ~)
A(**y**, ~, ~, ~, ~)
A(~, **y**, ~, ~, ~)
A(~, ~, **y**, ~, ~)
A(~, ~, ~, **y**, ~)

Contradiction! (with the weak one-wayness of f)

How do we prove the security of f’? By security reduction.

Suppose ∃ some PPT **A** th ability:

Takeaway:
R^A utilizes many independent copies of **A**!

f **A**

Then **R^A** inverts **f** with probability >3/4:

f **$R^{\boxed{A}}$**

Classically, we can write black-box reductions R^A:
R queries A many times

say we want to invert **y** = f(x):
A(~, ~, **y**, ~, ~)
A(**y**, ~, ~, ~, ~)
A(~, **y**, ~, ~, ~)
A(~, ~, **y**, ~, ~)
A(~, ~, ~, **y**, ~)

Contradiction! (with the weak one-wayness of f)

How do we prove the security of f'? By security reduction.

Suppose ∃ some PPT **A** th ability:

Takeaway:
$\mathbf{R}^{\mathbf{A}}$ utilizes many independent copies of **A**!

This is possible because we model **A** as an algorithm, which can be copied and run again.

Then $\mathbf{R}^{A}$ inverts **f** with prob

Classically, we can write black-box reductions $\mathbf{R}^{A}$:
R queries A many times

$$\mathbf{f} \rightleftarrows \mathbf{R}^{\boxed{\mathbf{A}}}$$

say we want to invert **y** = f(x):

A(~, ~, **y**, ~, ~)

A(**y**, ~, ~, ~, ~)

A(~, **y**, ~, ~, ~)

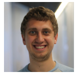
A(~, ~, **y**, ~, ~)

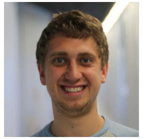
A(~, ~, ~, **y**, ~)

Contradiction! (with the weak one-wayness of f)

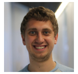
What if we can’t run A many times?

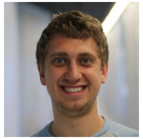
What if we can't run A many times?

Maybe **A** is your **"next door neighbor"**  who happens to break **f'**:

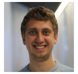
f' $\rightleftarrows$ 

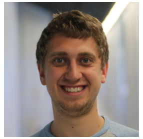
Cody

But you only have **"interactive access"** to  when trying to break **f**:

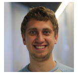
f $\rightleftarrows$ **R** $\rightleftarrows$ 

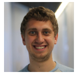
Suppose A can only be accessed interactively.

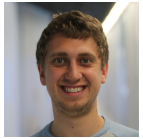
Maybe **A** is your **“next door neighbor”**  who happens to break **f’**:

f’ $\rightleftarrows$ 

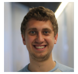
Cody

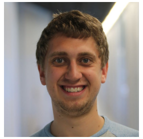
You have no clue how  **works.**

But you only have **“interactive access”** to  when trying to break **f:**

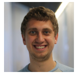
f $\rightleftarrows$ **R** $\rightleftarrows$ 

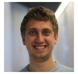
Suppose A can only be accessed interactively.

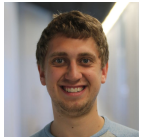
Maybe **A** is your **"next door neighbor"**  who happens to break **f'**:

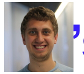
$$f' \rightleftarrows$$ 

Cody

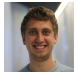
You have no clue how  works.

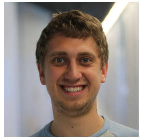
But you only have **"interactive access"** to  when trying to break **f:**

$$f \rightleftarrows R \rightleftarrows$$ 

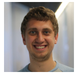
You don't know 's code…

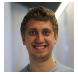
Suppose A can only be accessed interactively.

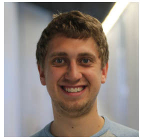
Maybe **A** is your **"next door neighbor"**  who happens to break **f'**:

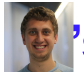
$$f' \rightleftarrows$$ 

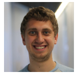
Cody

You have no clue how  works.

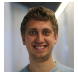
But you only have **"interactive access"** to  when trying to break **f**:

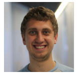
$$f \rightleftarrows R \rightleftarrows$$ 

You don't know 's code…

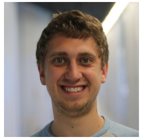
You can't "copy" 

Suppose A can only be accessed interactively.

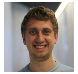
No “rewinding” 

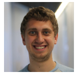
Maybe **A** is your **“next door neighbor”**  who happens to break **f’**:

$$f' \rightleftarrows$$

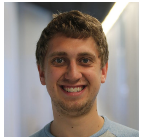


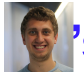
Cody

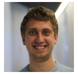
You have no clue how  **works.**

But you only have **“interactive access”** to  when trying to break **f:**

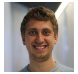
$$f \rightleftarrows R \rightleftarrows$$

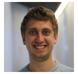


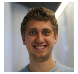
You don’t know  **’s code…**

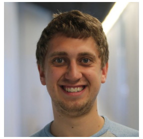
You can’t “copy” 

Suppose A can only be accessed interactively.

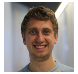
No “rewinding” 

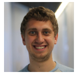
Maybe A is your **“next door neighbor”**  who happens to break **f’**:

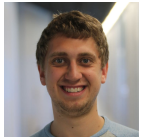
 **might have access to “cosmic resources” as far as you’re concerned**

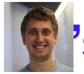
⇄ 

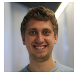
Cody

You have no clue how  **works.**

But you only have **“interactive access”** to  when trying to break **f**:

f ⇄ **R** ⇄ 

You don’t know  **’s code…**

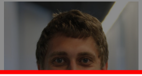
You can’t “copy” 

Suppose A can only be accessed interactively.

No "rewinding"

Maybe A is your **"next door neighbor"** who happens to break **f'**:

might have access to "cosmic resources" as f you're concer



You have no clue how A rks.

Claim: we need to revisit classical proofs!

But you only have **"interactive access"** to when trying to break **f**:

f ⇄ **R** ⇄

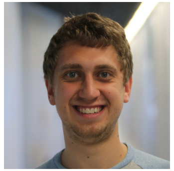
You don't know 's code…

You can't "copy"

Looking forward, even just sending Cody multiple queries…

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)

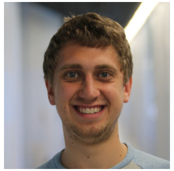


Cody

Looking forward, even just sending Cody multiple queries…

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)



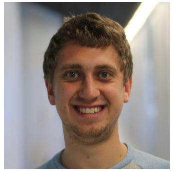
Cody

…might break down, since Cody is stateful.

Looking forward, even just sending Cody multiple queries…

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)



Cody

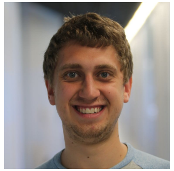
…might break down, since Cody is stateful.

A stateful adversary will remember that they've already answered a query.
"That's enough winning for today!"

Looking forward, even just sending Cody multiple queries…

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)



Cody

…might break down, since Cody is stateful.

Looking forward, we will assume that the adversary *wins "repeatedly" when given fresh challenges*. But even this is non-trivial to exploit.

“Stateful attackers” are already well motivated:

Quantum computers break existing proof techniques:

- No-cloning theorem: cannot copy quantum advice.
- Can’t be “rewound” when playing interactive security games

“Stateful attackers” are already well motivated:

Quantum computers break existing proof techniques:

- No-cloning theorem: cannot copy quantum advice.
- Can’t be “rewound” when playing interactive security games

Theoretically:

- We prefer a theory of cryptography that makes as few assumptions as possible!
- Can we get by without assuming that attackers are PPT (or QPT)?

This Talk:

We propose a reduction-based
theory of computational cryptography
with minimal assumptions on
the Nature of real-world attackers.

Next up: Defining Universal Reductions

After that: Feasibility and Impossibility Results

Defining Universal Reductions

A $\leftrightarrow$ Nat

"Attacker"
uniform **PPT**

"Nature"
nonuniform **any choice of runtime**
Stateful

"Augmented Adversary" (A, Nat)

A new model of attacker: “Augmented Adversaries”

A $\leftrightarrow$ Nat

“Attacker”
uniform **PPT**

“Nature”
nonuniform **any choice of runtime**
Stateful

“Augmented Adversary” (A, Nat)

Some unknown power in the cosmos to which we only have “interactive access”

A new model of attacker: “Augmented Adversaries”

A classic attacker
that uses Nature
to break some scheme

A ⟷ **Nat**

“Attacker”
uniform **PPT**

“Nature”
nonuniform **any choice of runtime**
Stateful

“Augmented Adversary” (A, Nat)

Some unknown power in the cosmos to which we only have “interactive access”

Augmented Security Game

$$\mathsf{C} \leftrightarrow \mathsf{A} \leftrightarrow \mathsf{Nat}$$

"Challenger"
uniform **PPT**
outputs "win"/"lose"

"Attacker"
uniform **PPT**

"Nature"
nonuniform **any choice of runtime**

Augmented Security Game

$$\mathbf{C} \leftrightarrow \mathbf{A} \leftrightarrow \mathbf{Nat}$$

"Challenger"
uniform **PPT**
outputs "win"/"lose"

"Attacker"
uniform **PPT**

"Nature"
nonuniform **any choice of runtime**

***Observe*: the attacker can alter the state of Nature during the interaction. This is intentional and a key property of our definition.**

Note: all communication is classical (and **C**/**A** are PPT) because we want universal reductions to work in a PPT world!

Robust winning: “winning repeatedly”

Recall: We want adversaries that win “repeatedly” when given fresh challenges.

$$\mathbf{C} \leftrightarrow \mathbf{A} \leftrightarrow \mathbf{Nat}(\rho)$$

“Challenger”
uniform **PPT**
outputs “win”/“lose”

“Attacker”
uniform **PPT**

“Nature”
nonuniform **any choice of runtime**

Interaction prefix ρ:
a transcript of messages previously sent to Nat before the beginning of execution, including coins flipped by Nat.

Robust winning: “winning repeatedly”

$$\mathbf{C} \leftrightarrow \mathbf{A} \leftrightarrow \mathbf{Nat}(\rho)$$

“Challenger”
uniform **PPT**
outputs “win”/“lose”

“Attacker”
uniform **PPT**

“Nature”
nonuniform **any choice of runtime**

Definition: (A, Nat) has **robust advantage** a(.) for C, if ∀ *interaction prefixes* ρ, ∀ λ:
Pr[(**A**, **Nat**(ρ)) wins **C**] ≥ a(λ)

Interaction prefix ρ:
a transcript of messages previously sent to Nat before the beginning of execution, including coins flipped by Nat.

Universal Reductions

∃ an **ε-universal reduction** from **C** to **C'** if ∀ PPT **A**, ∃ PPT **A'** s.t. ∀**Nat**:

Suppose (**A**, **Nat**) has **robust advantage a(·)** for **C**

$$\begin{tikzcd}
C \arrow[r, leftrightarrow, shift left=2] \arrow[r] & A \arrow[l, shift left=2] \arrow[r, leftrightarrow] & Nat
\end{tikzcd}$$

Then (**A'**, **Nat**) has **robust advantage ε(·,a(·))** for **C'**.

$$\begin{tikzcd}
C' \arrow[r, shift left, "f(x)"] & A' \arrow[l, shift left, "f^{-1}(f(x))"] \arrow[r, leftrightarrow] & Nat
\end{tikzcd}$$

Universal Reductions

This is the central notion in our paper.

$\exists$ an **ε-universal reduction** from **C** to **C'** if $\forall$ PPT **A**, $\exists$ PPT **A'** s.t. $\forall$ **Nat**:

Suppose (**A**, **Nat**) has **robust advantage a(·)** for **C**

$$\begin{tikzcd} \mathbf{C} \arrow[r, leftrightarrow, shift left=3] \arrow[r, shift left=0] \arrow[r, leftarrow, shift right=3] & \mathbf{A} \arrow[r, leftrightarrow] & \mathbf{Nat} \end{tikzcd}$$

Then (**A'**, **Nat**) has **robust advantage ε(·,a(·))** for **C'**.

A' can now use the fact that (A, Nat) wins "on demand".

$$\begin{tikzcd} \mathbf{C'} \arrow[r, shift left, "f(x)"] & \mathbf{A'} \arrow[l, shift left, "f^{-1}(f(x))"] \arrow[r, leftrightarrow] & \mathbf{Nat} \end{tikzcd}$$

Universal Reductions

This is the central notion in our paper.

∃ an **ε-universal reduction** from **C** to **C'** if ∀ PPT **A**, ∃ PPT **A'** s.t. ∀**Nat**:

Suppose (**A**, **Nat**) has **robust advantage a(·)** for **C**

$$\begin{tikzcd} \mathbf{C} \arrow[r, leftrightarrow, shift left=3] \arrow[r] & \mathbf{A} \arrow[l, shift left=3] \arrow[r, leftrightarrow] & \mathbf{Nat} \end{tikzcd}$$

Then (**A'**, **Nat**) has **robust advantage ε(·,a(·))** for **C'**.

A' can now use the fact that (A, Nat) wins "on demand".

$$\begin{tikzcd} \mathbf{C'} \arrow[r, shift left, "f(x)"] & \mathbf{A'} \arrow[l, shift left, "f^{-1}(f(x))"] \arrow[r, leftrightarrow] & \mathbf{Nat} \end{tikzcd}$$

Are Universal Reductions universal? Certainly, universal reductions imply reductions w.r.t. (nu)PPT, (nu)QPT.

Quick Comparisons

Relativized Reductions

- A *relativized reduction* gives attackers A^O access to some arbitrary oracle O
- O is modeled as a (perhaps uncomputable) function
- Universal reductions can be viewed as *relativized reductions* for *stateful, interactive oracles* O (in contrast to a *non-interactive*, *stateless* oracle).

Universal Composability [Canetti00]

- Universal reductions are *syntactically* similar to UC with unbounded environments
- *Semantically* very different: our notion is reduction-based & computational. (For instance, UC *security proofs* can rewind the environment [e.g. CLP10])

What can we do with universal reductions?

Warmup: a basic feasibility result

Thm 1. Classical 1-shot straight-line black-box reductions imply universal reductions.

> A straightforward argument, since a 1-shot reduction uses Nature once.

> **Corollaries**: Witness Indistinguishability/PRG Length Extension/PRFs/SKE/Commitments from PRGs

Warmup: a basic feasibility result

Thm 1. Classical 1-shot straight-line black-box reductions imply universal reductions.

> A straightforward argument, since a 1-shot reduction uses Nature once.

> **Corollaries**: Witness Indistinguishability/PRG Length Extension/PRFs/SKE/Commitments from PRGs

What about problems that have classical reductions invoking the attacker multiple times?

Unfortunately, not all is possible...

<u>Thm 2</u> (*Impossibility of Hardness Amplification*):
There is no universal black-box reduction from the OWF security of $g^n(x_1 \ldots x_n) = (g(x_1), \ldots, g(x_n))$ to the OWF security of $g(x)$ that uses only black-box access to g, and that works for any function g.

<u>Thm 3</u> (*Impossibility of a Goldreich-Levin-Style Theorem*):
There is no universal black-box reduction from the security of the hardcore predicate $h(x,r) = \langle x,r \rangle$ w.r.t. $f(x, r) = (g(x), r)$ to the OWF security of g that uses only black-box access to g and that works for any function g.

Unfortunately, not all is possible...

<u>Thm 2</u> (*Impossibility of Hardness Amplification*):
There is no universal black-box reduction from the OWF security of $g^n(x_1 \ldots x_n) = (g(x_1), \ldots, g(x_n))$ to the OWF security of $g(x)$ that uses only black-box access to g, and that works for any function g.

<u>Thm 3</u> (*Impossibility of a Goldreich-Levin-Style Theorem*):
There is no universal black-box reduction from the security of the hardcore predicate $h(x,r) = \langle x,r \rangle$ w.r.t. $f(x, r) = (g(x), r)$ to the OWF security of g that uses only black-box access to g and that works for any function g.

Let's go over the intuition of the proofs to understand universal reductions better.

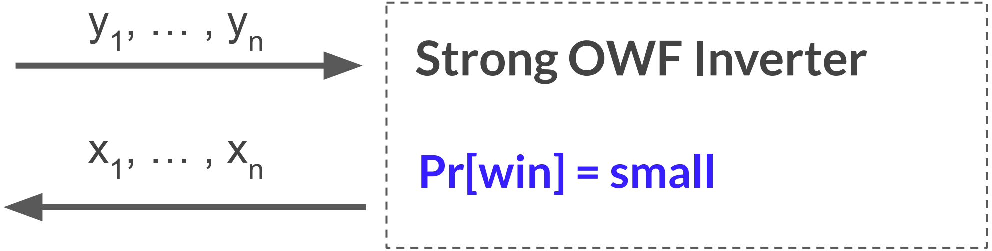
Recall: classical Hardness Amplification

Let g be a weak OWF. Then $g^n = (g(x_1), \dots, g(x_n))$ is a strong OWF.

Recall: classical Hardness Amplification

Let g be a weak OWF. Then $g^n = (g(x_1), \ldots, g(x_n))$ is a strong OWF.

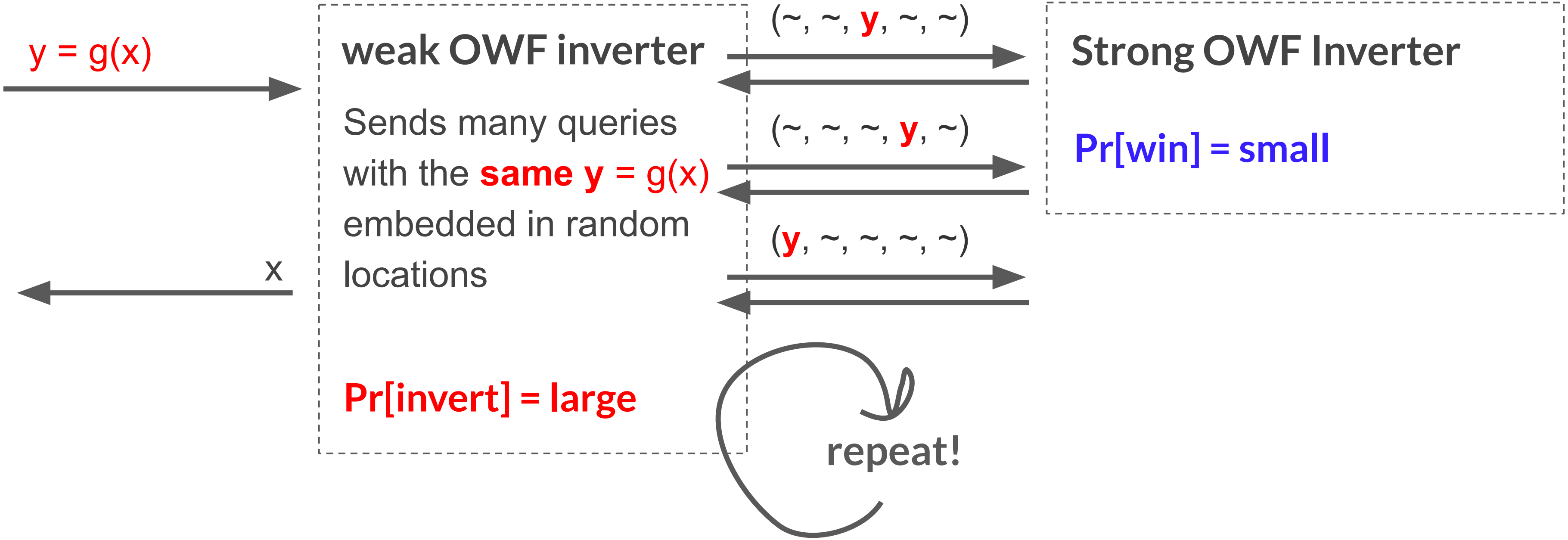
Proof: Suppose g^n is not a strong OWF…



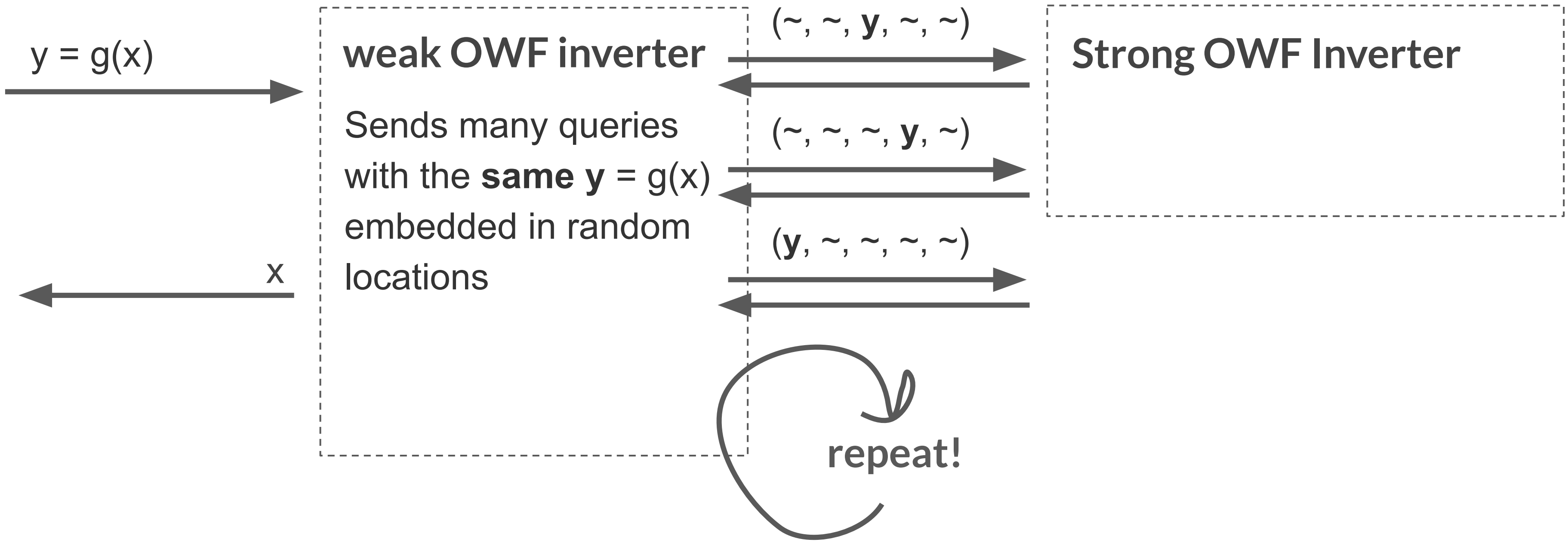
Recall: classical Hardness Amplification

Let g be a weak OWF. Then $g^n = (g(x_1), \dots, g(x_n))$ is a strong OWF.

Proof: Suppose g^n is not a strong OWF…

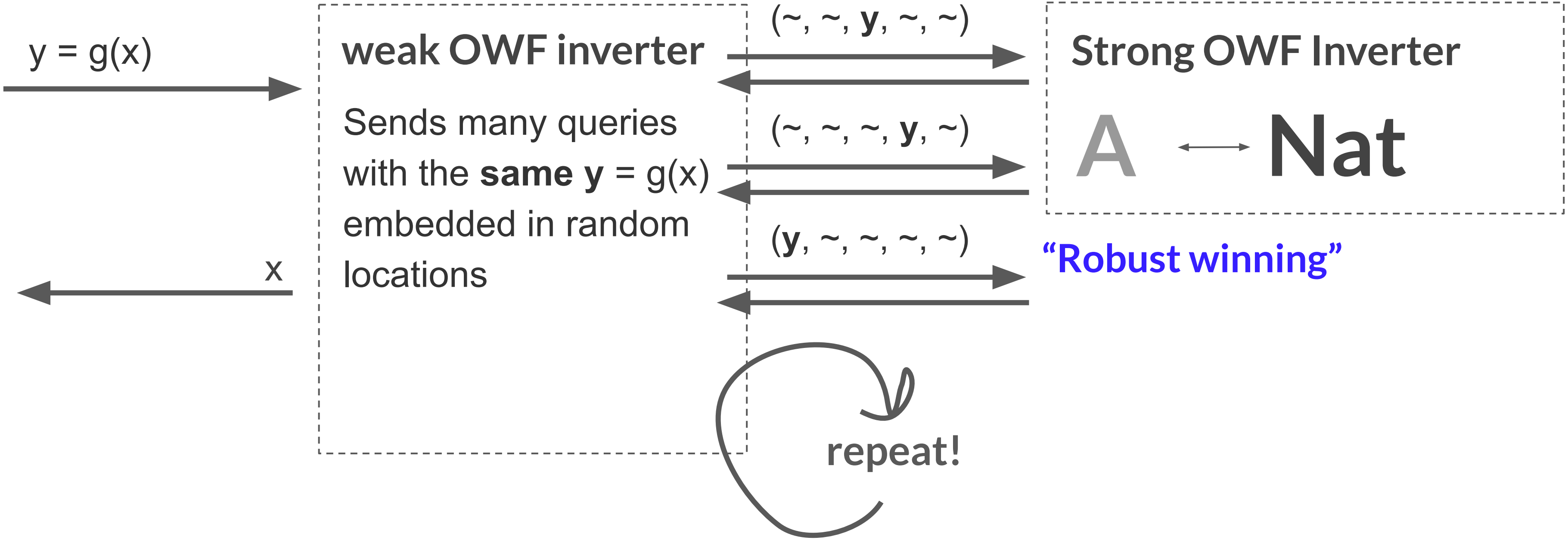


Can we write a universal reduction?



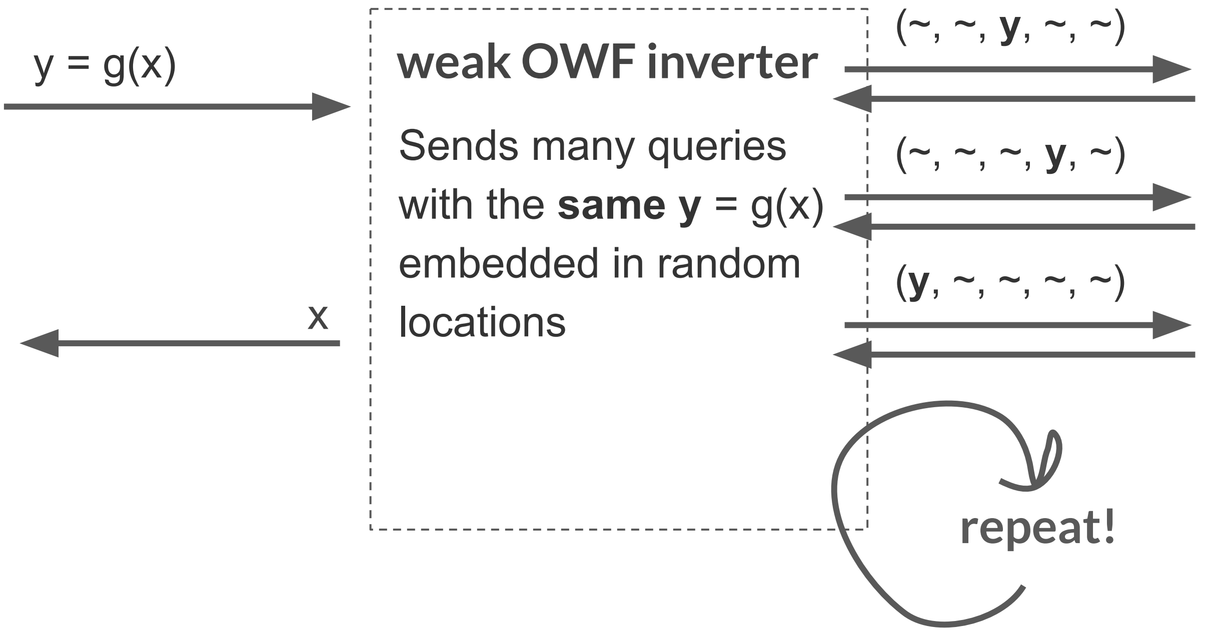
Can we write a universal reduction?

Suppose g^n is not a strong OWF…



Can we write a universal reduction?

Suppose g^n is not a strong OWF…



Consider the following (**A**, **Nat**):

- **A** forwards messages
- **Nat** keeps a list ***Seen*** of previously processed subqueries y_i
- **Nat** on seeing $(y_1, y_2, \ldots, y_n)$,
  - If any $y_i \in$ ***Seen***, **Nat** replies "reject"
  - Else, **Nat** replies with correct preimage with desired Pr.
  - Finally, add each y_i to ***Seen***

Can we write a universal reduction?

Suppose g^n is not a strong OWF…

Since Nat is stateful,
if it previously saw y, *it can ignore future correlated queries (e.g. that contain* ***y****.)*

$y = g(x)$

weak OWF inverter

Sends many queries with the **same y** = g(x) embedded in random locations

x

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)

(**y**, ~, ~, ~, ~)

repeat!

Strong OWF Inverter

A ⟷ **Nat**

Consider the following (**A**, **Nat**):

- **A** forwards messages
- **Nat** keeps a list ***Seen*** of previously processed subqueries y_i
- **Nat** on seeing $(y_1, y_2, \ldots, y_n)$,
  - If any $y_i \in$ ***Seen***, **Nat** replies "reject"
  - Else, **Nat** replies with correct preimage with desired Pr.
  - Finally, add each y_i to ***Seen***

Can we write a universal reduction?

Since Nat is stateful,
if it previously saw y, *it can ignore future correlated queries (e.g. that contain* ***y****.)*

Suppose g^n is not a strong OWF…

$y = g(x)$

weak OWF inverter

Sends many queries with the **same y** = g(x) embedded in random locations

Only the first query will be useful!

x

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)

(**y**, ~, ~, ~, ~)

repeat!

Strong OWF Inverter

A ⟷ **Nat**

Consider the following (**A**, **Nat**):

- **A** forwards messages
- **Nat** keeps a list ***Seen*** of previously processed subqueries y_i
- **Nat** on seeing $(y_1, y_2, \ldots, y_n)$,
  - If any $y_i \in$ ***Seen***, **Nat** replies "reject"
  - Else, **Nat** replies with correct preimage with desired Pr.
  - Finally, add each y_i to ***Seen***

Can we write a universal reduction?

Suppose g^n is not a strong OWF…

Observe:
(A, Nat) is still robustly winning!
Probability a fresh challenge coincides with "Seen" strings is tiny

$y = g(x)$

weak OWF inverter

Sends many queries with the **same y** = g(x) embedded in random locations

x

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)

(**y**, ~, ~, ~, ~)

Only the first query will be useful!

repeat!

Strong OWF Inverter

A ⟷ **Nat**

Consider the following (**A**, **Nat**):

- **A** forwards messages
- **Nat** keeps a list ***Seen*** of previously processed subqueries y_i
- **Nat** on seeing $(y_1, y_2, \ldots, y_n)$,
  - If any $y_i \in$ ***Seen***, **Nat** replies "reject"
  - Else, **Nat** replies with correct preimage with desired Pr.
  - Finally, add each y_i to ***Seen***

Can we write a universal reduction?

Observe:
(A, Nat) is still robustly winning!
Probability a fresh challenge coincides with "Seen" strings is tiny

Suppose g^n is not a strong OWF…

$y = g(x)$

weak OWF inverter

Sends many queries with the **same y** = g(x) embedded in random locations

x

(~, ~, **y**, ~, ~)

(~, ~, ~, **y**, ~)

(**y**, ~, ~, ~, ~)

Strong OWF Inverter

A ⟷ **Nat**

repeat!

In the full proof, we show that any reduction – not just Yao's– that is black-box in g will have to send correlated queries to (**A**, **Nat**).
Thus, similarly broken!

Consider the following (**A**, **Nat**):

- **A** forwards messages
- **Nat** keeps a list ***Seen*** of previously processed subqueries y_i
- **Nat** on seeing $(y_1, y_2, \ldots, y_n)$,
  - If any $y_i \in$ ***Seen***, **Nat** replies "reject"
  - Else, **Nat** replies with correct preimage with desired Pr.
  - Finally, add each y_i to ***Seen***

Indeed,
Hardness amplification is possible for specific one-way functions!

Theorem 4 (Informal):
Let f be a re-randomizable OWF. Then Yao's reduction is a universal reduction.

> Rerandomizability helps us fool Nature into thinking that it is always playing "fresh instances" of the security game.

Indeed,
Hardness amplification is possible for specific one-way functions!

Theorem 4 (Informal):
Let f be a re-randomizable OWF. Then Yao's reduction is a universal reduction.

> Rerandomizability helps us fool Nature into thinking that it is always playing "fresh instances" of the security game.

Writing universal reductions requires new techniques!

Indeed,
Hardness amplification is possible for specific one-way functions!

Theorem 4 (Informal):
Let f be a re-randomizable OWF. Then Yao's reduction is a universal reduction.

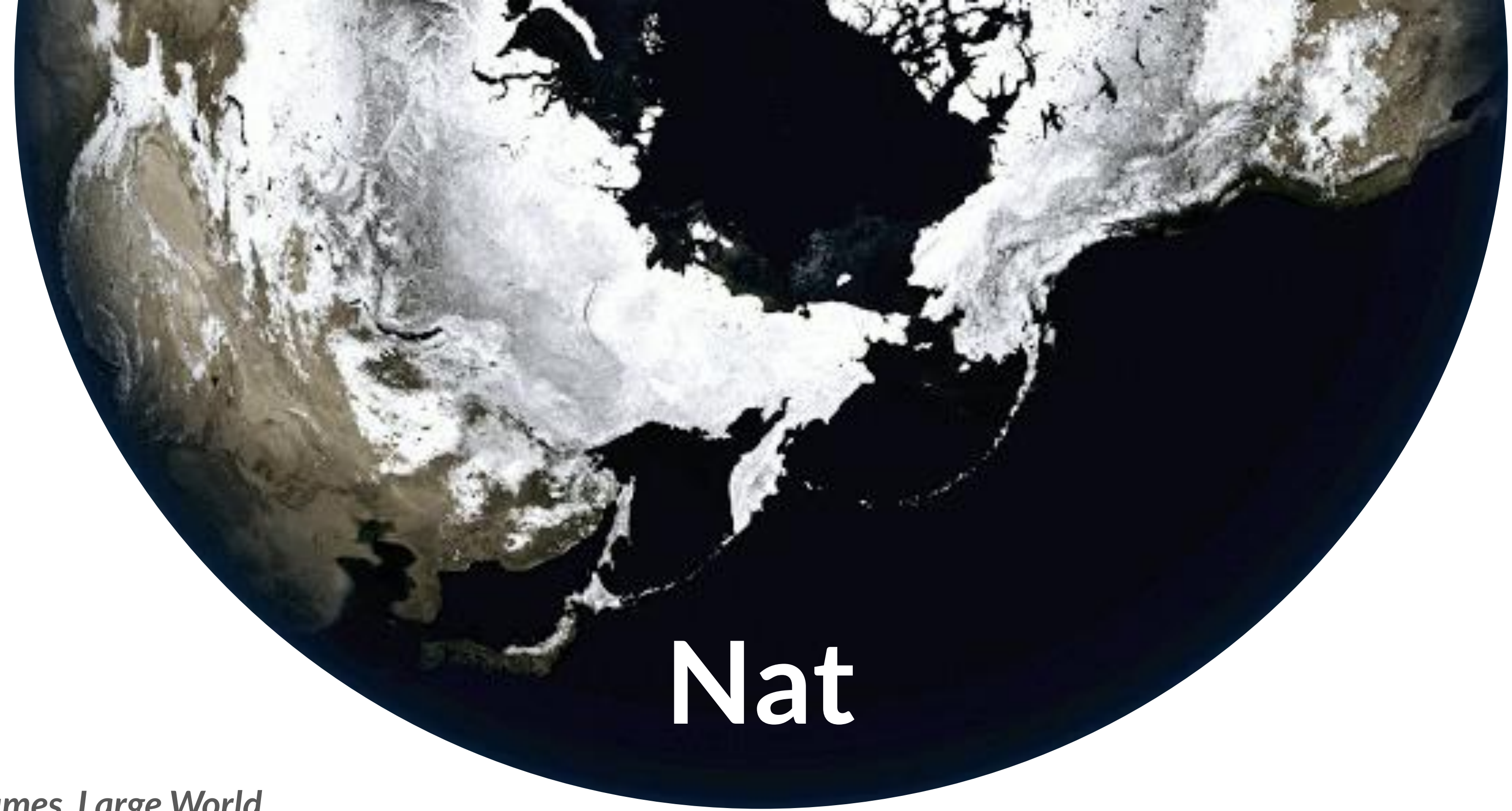
> Rerandomizability helps us fool Nature into thinking that it is always playing "fresh instances" of the security game.

Writing universal reductions requires new techniques!

for now, let's try to climb a different mountain…

Briefly: Restricting Nature

Can we get non-trivial results by imposing constraints on Nature?



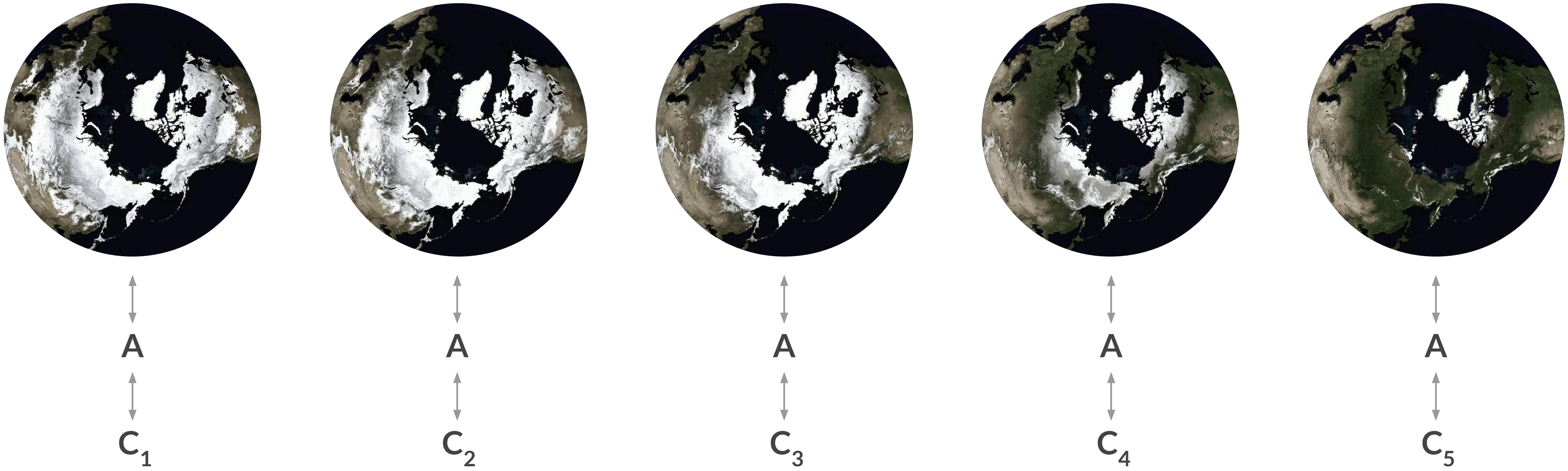
A

C

<u>***Small Games, Large World***</u>

It may be presumptuous to think that **C** or **A** can *influence* the future behavior of **Nat**.

What if **Nat** evolves over time (# of queries it has received)...



...but has a short term memory, and behaves independently of prior interactions?

Time-Evolving Windowed Natures

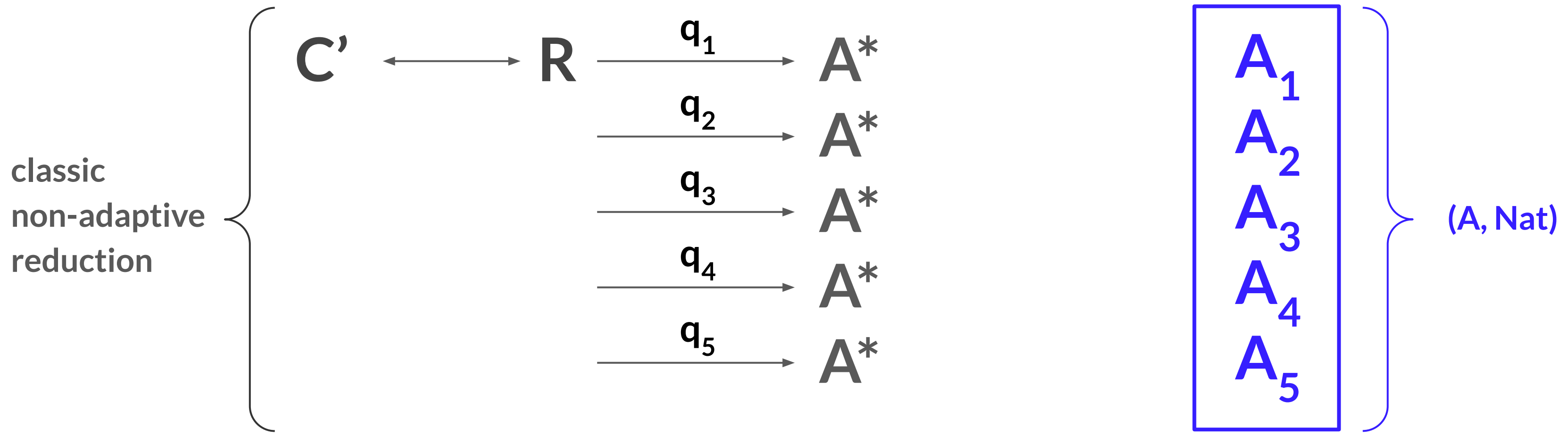
Theorem 5 (informal): Time-Evolving k-window Natures.
Suppose the behavior of "Nature" depends only on the number of messages it has seen, and the last k messages it has seen. Then classical non-adaptive straightline black-box reductions imply universal reductions w.r.t. this Nature.

We can think of (**A**, **Nat**) as a sequence of attackers $\mathbf{A_1}$ $\mathbf{A_2}$ $\mathbf{A_3}$ ...

How do we turn a "sequence of attackers" that must be queried in order into a single "restartable" adversary?

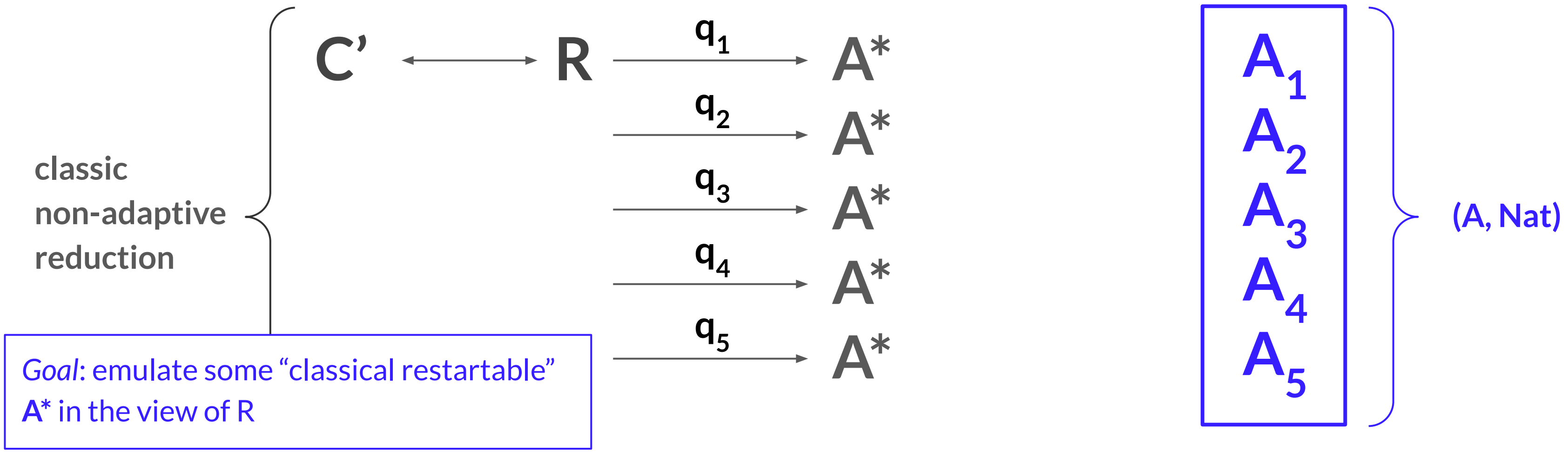
Time-Evolving Windowed Natures

Theorem 5 (informal): Time-Evolving k-window Natures.
Suppose the behavior of “Nature” depends only on the number of messages it has seen, and the last k messages it has seen. Then classical non-adaptive straightline black-box reductions imply universal reductions w.r.t. this Nature.



Time-Evolving Windowed Natures

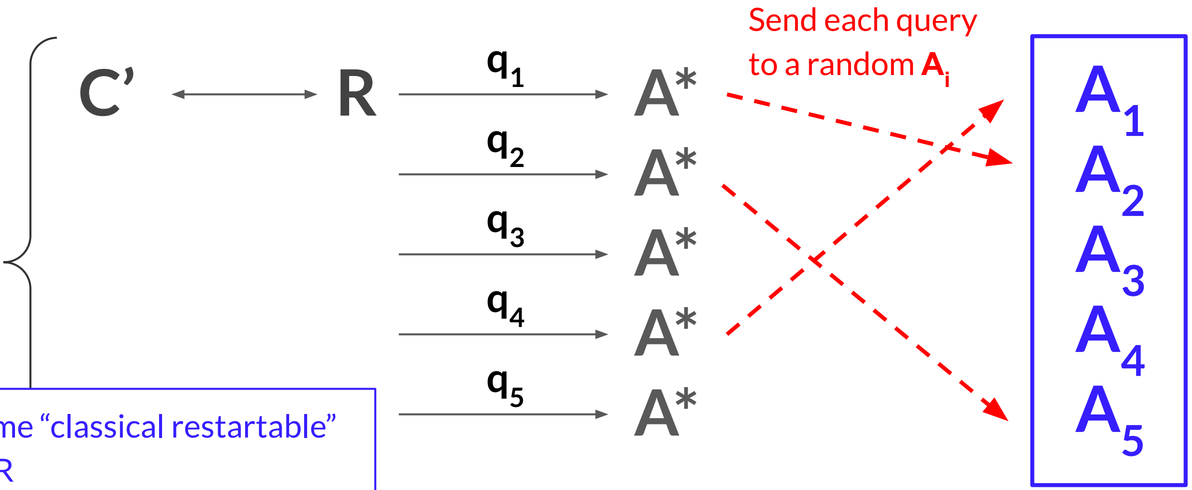
Theorem 5 (informal): Time-Evolving k-window Natures.
Suppose the behavior of "Nature" depends only on the number of messages it has seen, and the last k messages it has seen. Then classical non-adaptive straightline black-box reductions imply universal reductions w.r.t. this Nature.



Time-Evolving Windowed Natures

Theorem 5 (informal): Time-Evolving k-window Natures.
Suppose the behavior of "Nature" depends only on the number of messages it has seen, and the last k messages it has seen. Then classical non-adaptive straightline black-box reductions imply universal reductions w.r.t. this Nature.

classic non-adaptive reduction



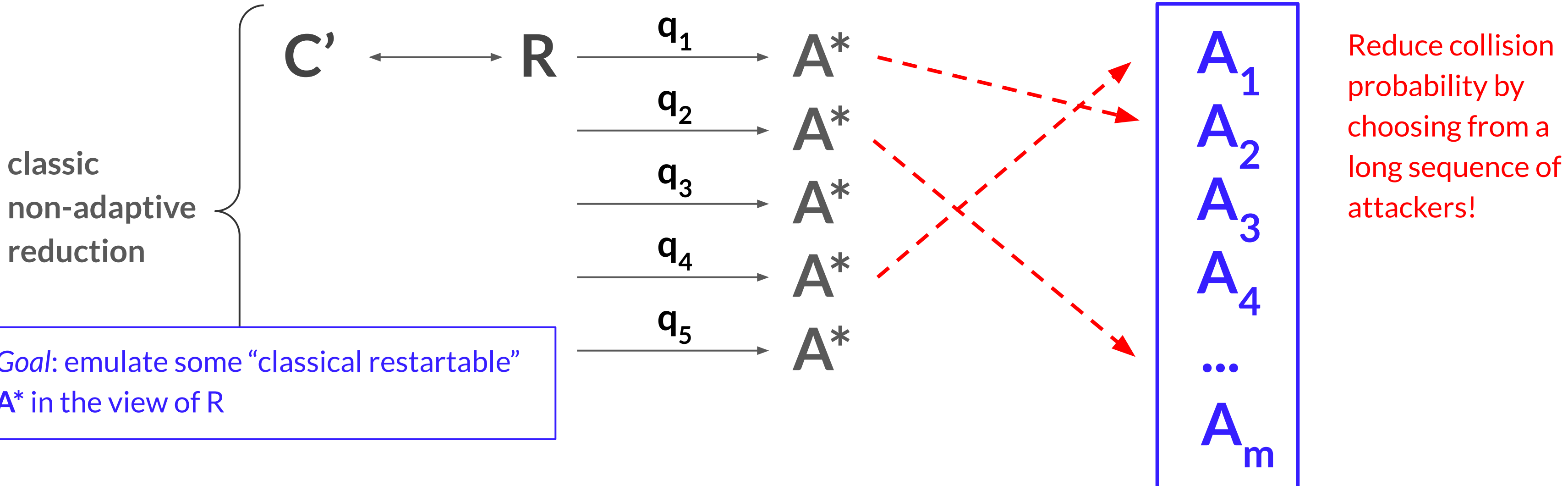
Nature is going to reply to queries out of order as it evolves $A_1, A_2, A_3 \ldots$

> Allowed since order of $q_1 \ldots q_5$ doesn't matter, by nonadaptivity.

Goal: emulate some "classical restartable" A* in the view of R

Time-Evolving Windowed Natures

Theorem 5 (informal): Time-Evolving k-window Natures.
Suppose the behavior of "Nature" depends only on the number of messages it has seen, and the last k messages it has seen. Then classical non-adaptive straightline black-box reductions imply universal reductions w.r.t. this Nature.



Goal: emulate some "classical restartable" A* in the view of R

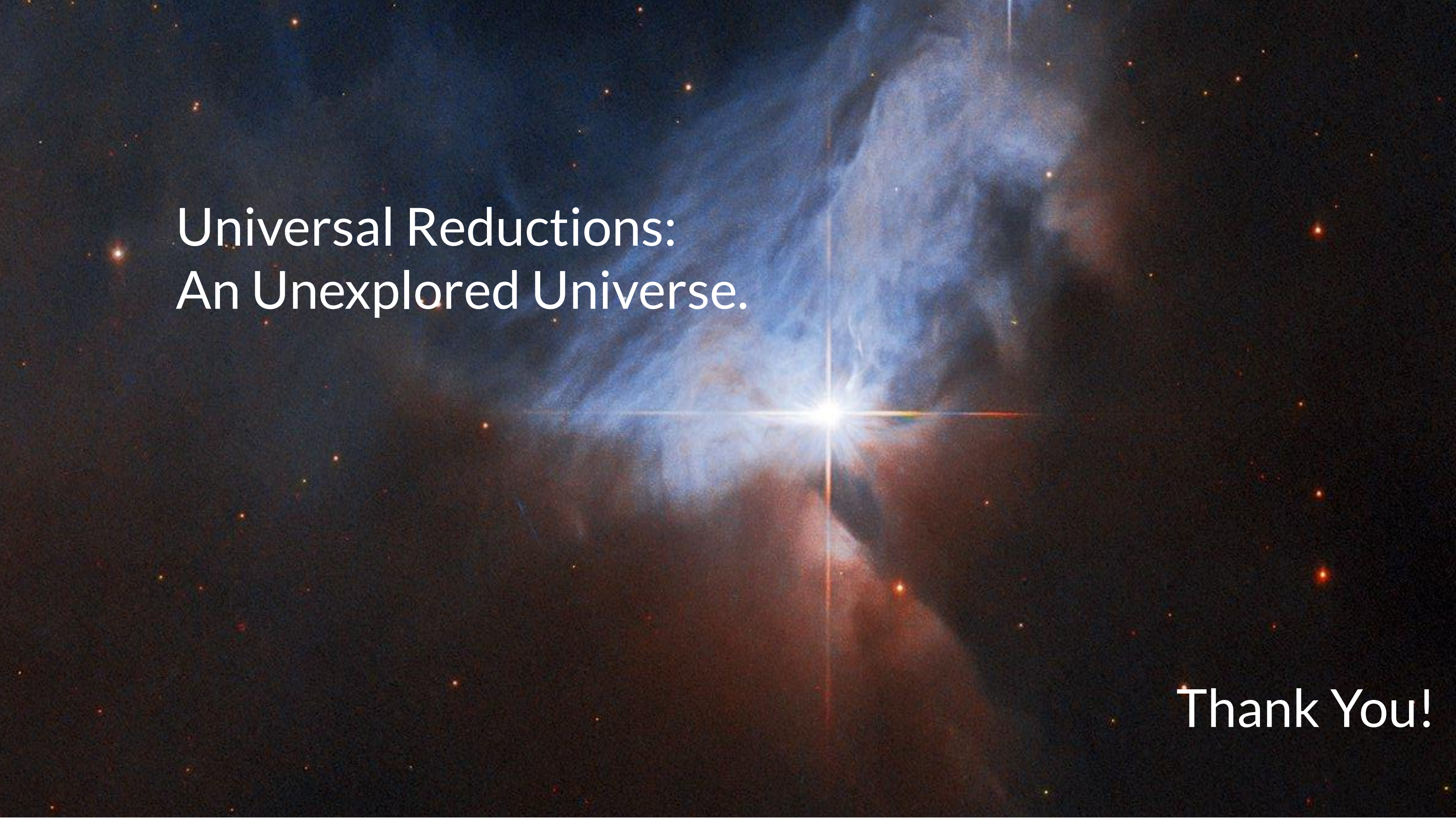
In conclusion: alot to unpack.

Takeaway: we can write meaningful security proofs w.r.t. stateful attackers!

Yet, at the same time, new techniques are clearly necessary.

- PRGs from OWFs?
- MPC?

We have hope for a “future-proof” notion of cryptography…


Universal Reductions:
An Unexplored Universe.
Thank You!