PPP-completeness with Connections to Cryptography

Manolis Zampetakis
MIT

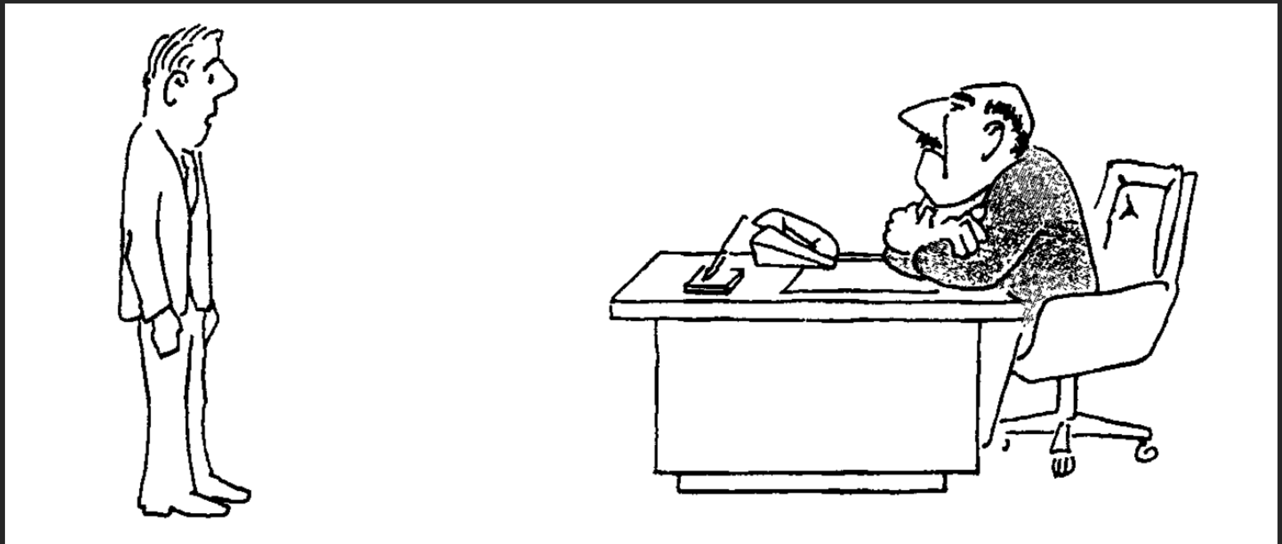
Katerina Sotiraki
MIT

Giorgos Zirdelis
Northeastern University

CoReLab Seminar 2020

Motivation via Cryptography

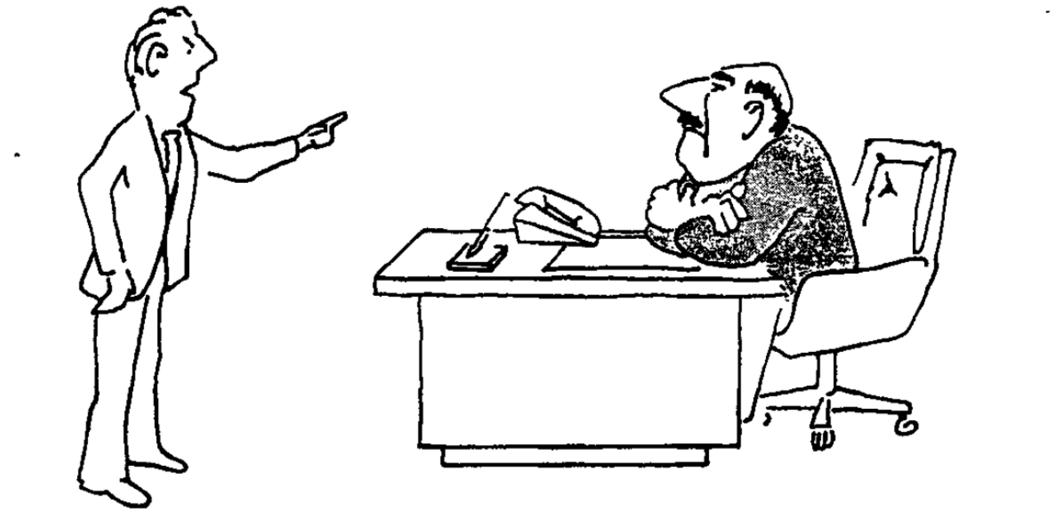
NP-hardness



"I can't find an efficient algorithm, I guess I'm just too dumb."

pictures from "Computers and Intractability" by Garey and Johnson 1979.

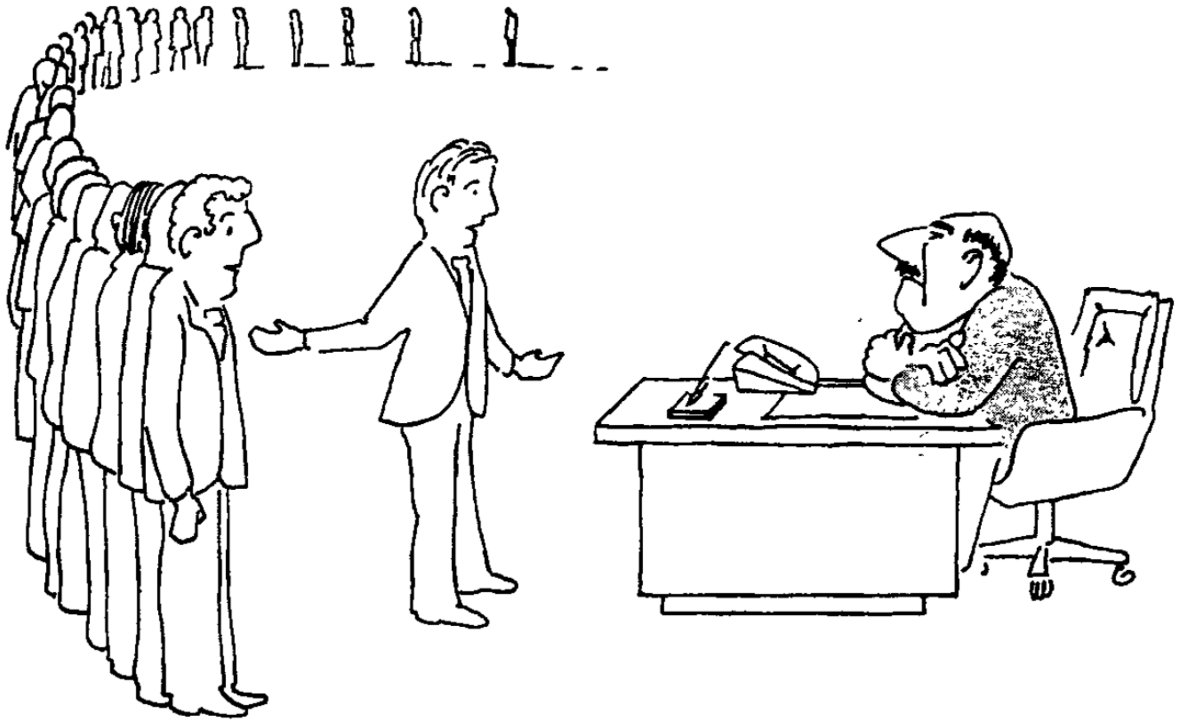
NP-hardness



"I can't find an efficient algorithm, because no such algorithm is possible!"

pictures from "Computers and Intractability" by Garey and Johnson 1979.

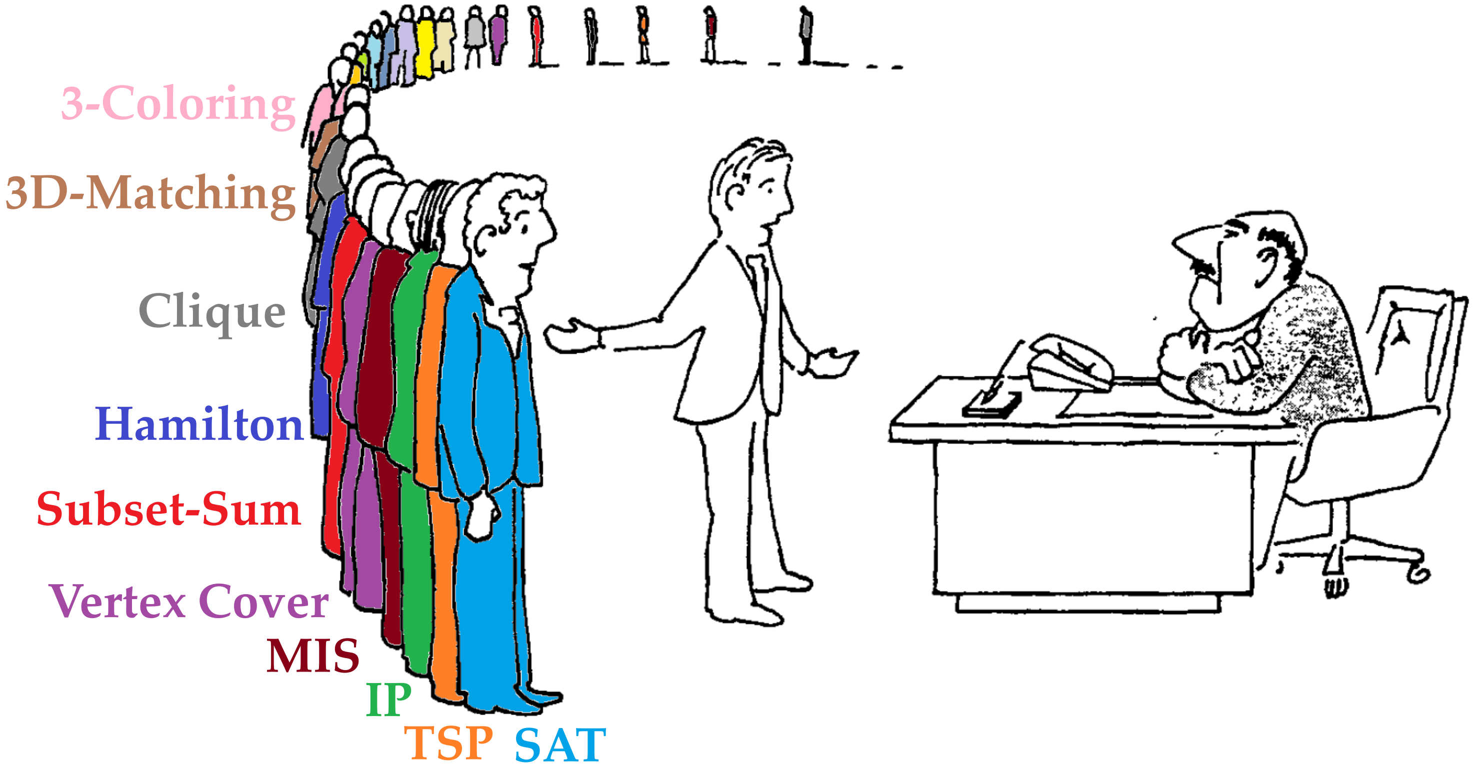
NP-hardness



“I can't find an efficient algorithm, but neither can all these famous people.”

pictures from “Computers and Intractability” by Garey and Johnson 1979.

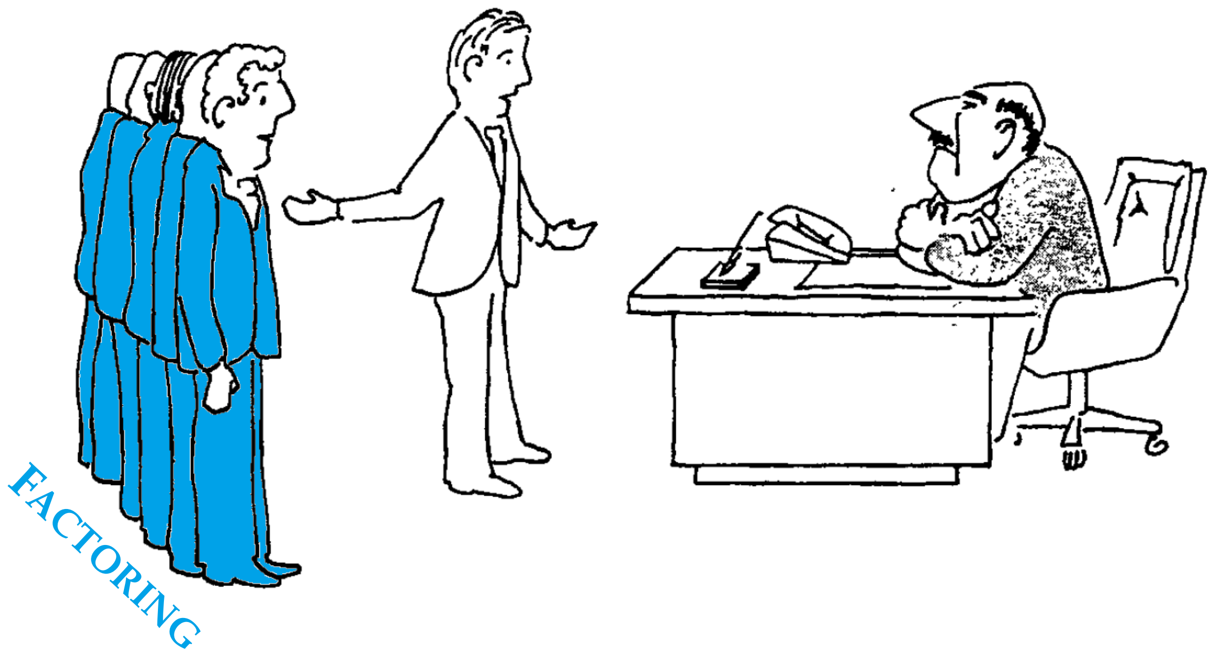
NP-hardness



"If I could find an algorithm I could solve all these famous difficult problems"

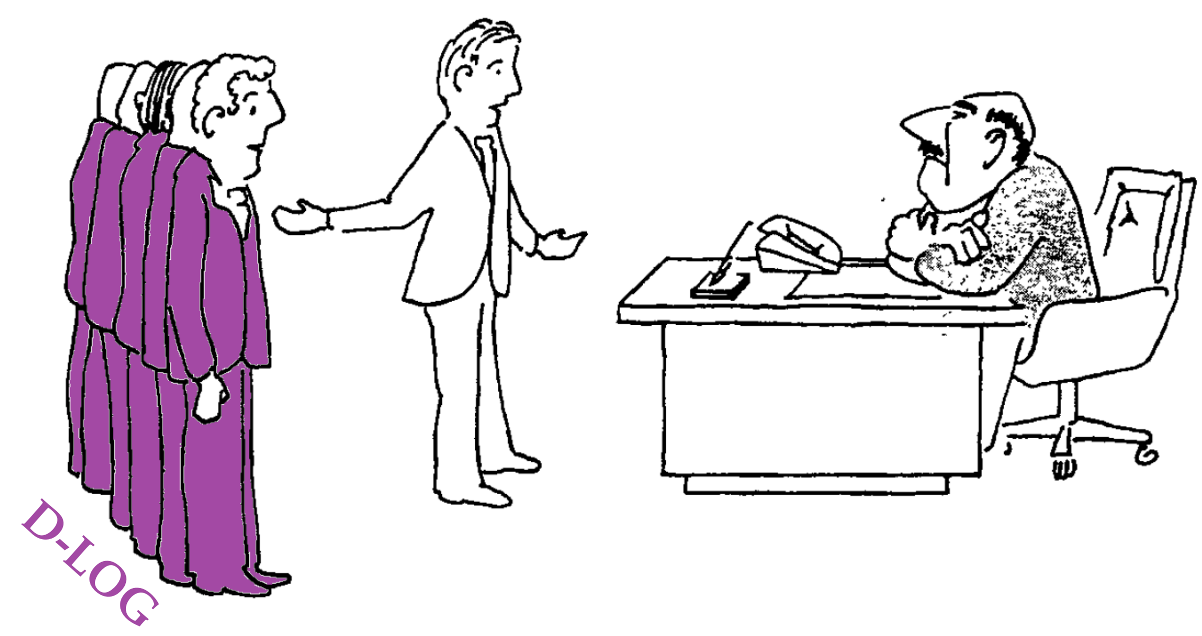
Cryptographic Hardness?

Cryptographic Hardness



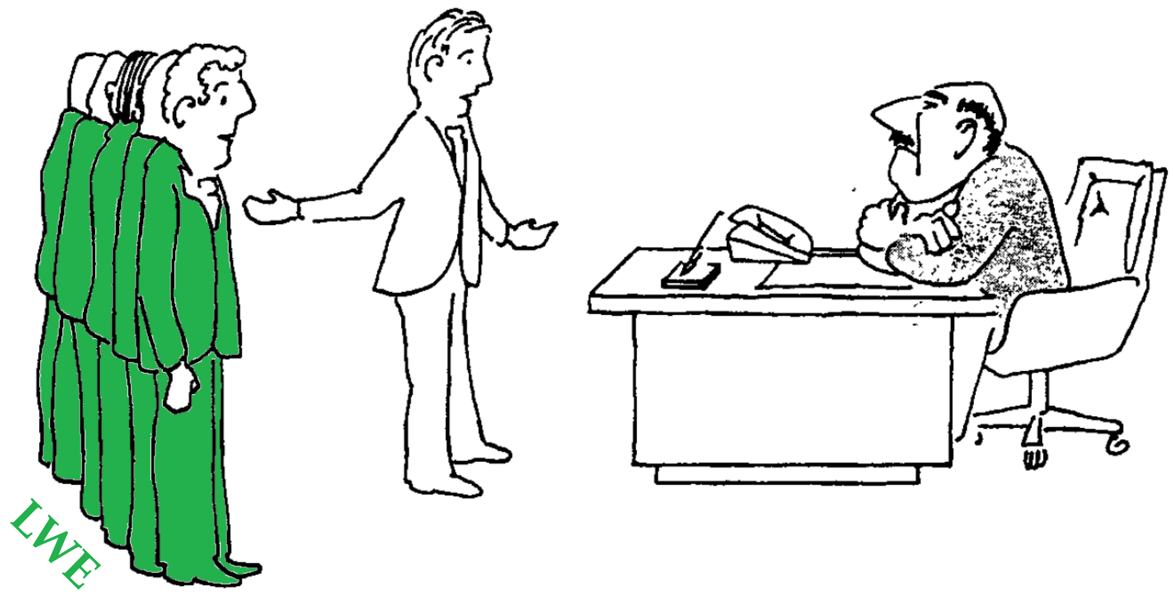
"If someone could break the protocol, they could solve **FACTORING** *on average*."

Cryptographic Hardness



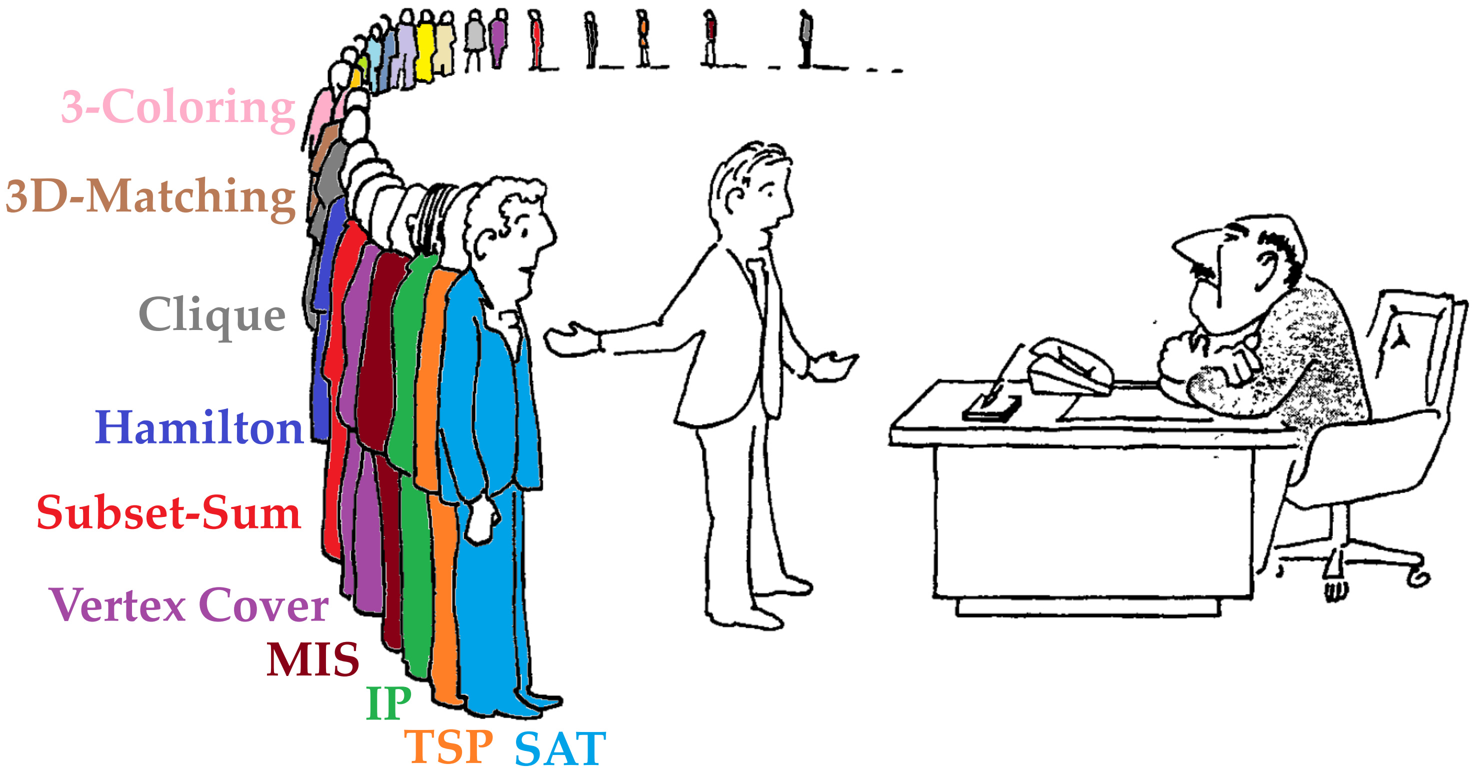
"If someone could break the protocol, they could solve **DISCRETE-LOG** *on average*."

Cryptographic Hardness



"If someone could break the protocol, they could solve **LWE** *on average*."

Utopia Cryptographic Hardness



"If someone could break the protocol, they could solve *on worst-case* all these famous difficult problems"

Can we achieve Cryptographic Utopia?

Can we achieve Cryptographic Utopia?

Bottlenecks

- cryptography is based on problems that are hard **on average!**

Can we achieve Cryptographic Utopia?

Bottlenecks

- cryptography is based on problems that are hard **on average!**
- NP-hard problems **do not suffice** for cryptography.

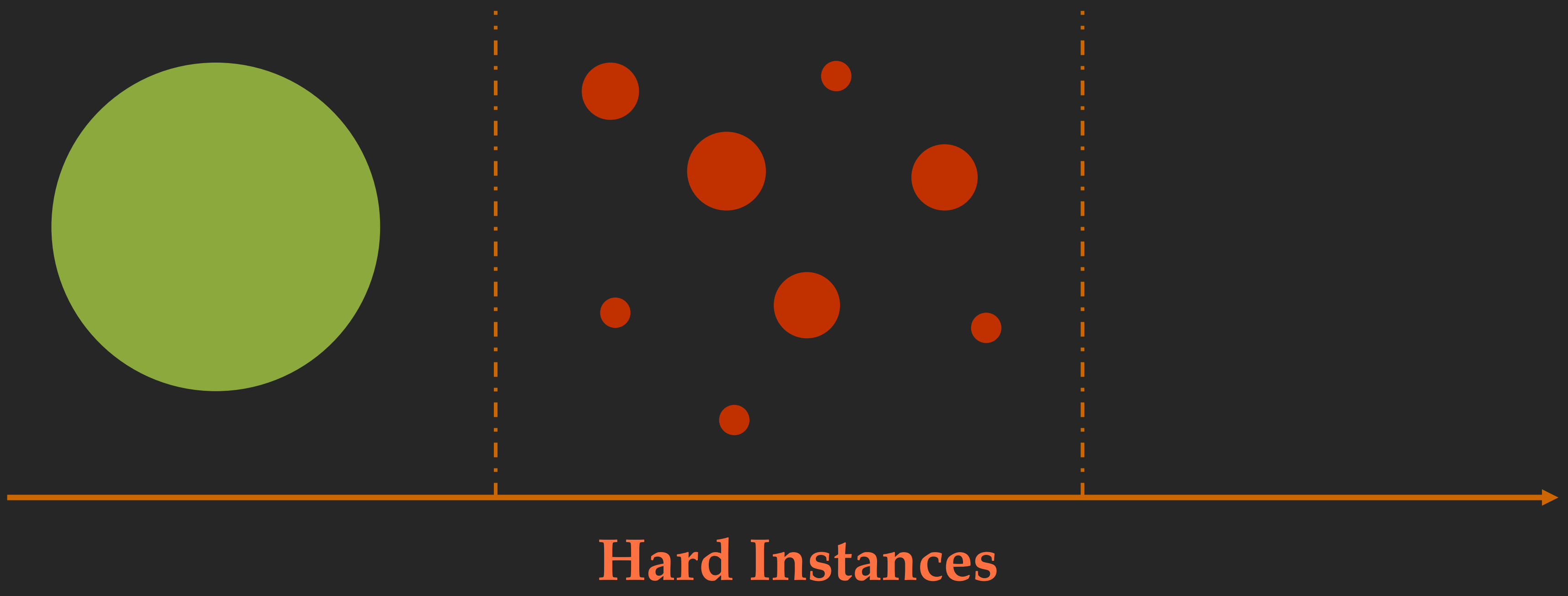
Can we achieve Cryptographic Utopia?

Bottlenecks

- **cryptography is based on problems that are hard on average!**

- NP-hard problems do not suffice for cryptography.

Average-Case Hardness



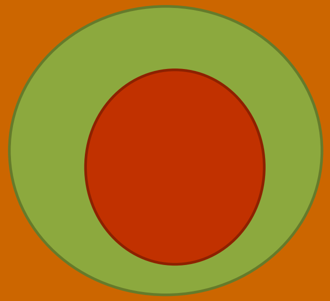
…but does not help for cryptographic utopia.

Worst-to-Average Case Reduction

Worst-to-Average Case Reduction

Average Case Hardness

Exists a distribution D over instances such that if we sample x from D, then x is hard with probability 0.5.

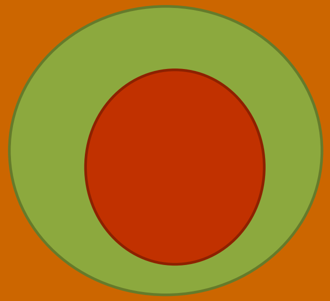


Worst-to-Average Case Reduction

worst-case problem
e.g. 3-SAT

Average Case Hardness

Exists a distribution D over instances such that if we sample x from D, then x is hard with probability 0.5.



Can we achieve Cryptographic Utopia?

Bottlenecks

- cryptography is based on problems that are hard on average!
- NP-hard problems do not suffice for cryptography.

Can we achieve Cryptographic Utopia?

Bottlenecks

- cryptography is based on problems that are hard on average!

we know problems that admit worst-to-average case reductions!

- NP-hard problems do not suffice for cryptography.

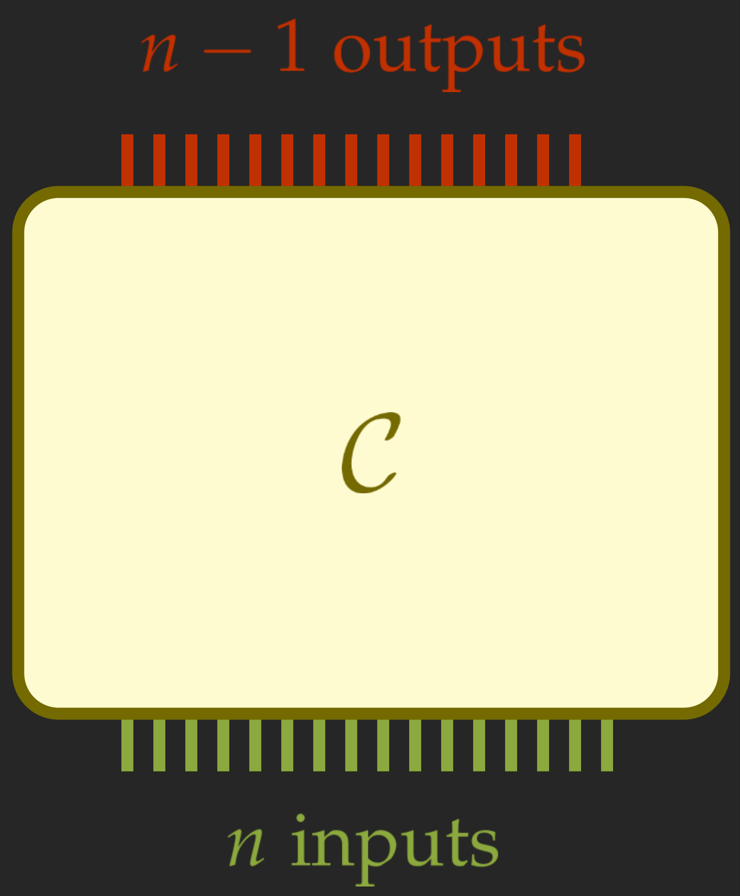
Can we achieve Cryptographic Utopia?

Bottlenecks

- cryptography is based on problems that are hard on average!
- **NP-hard problems do not suffice for cryptography.**

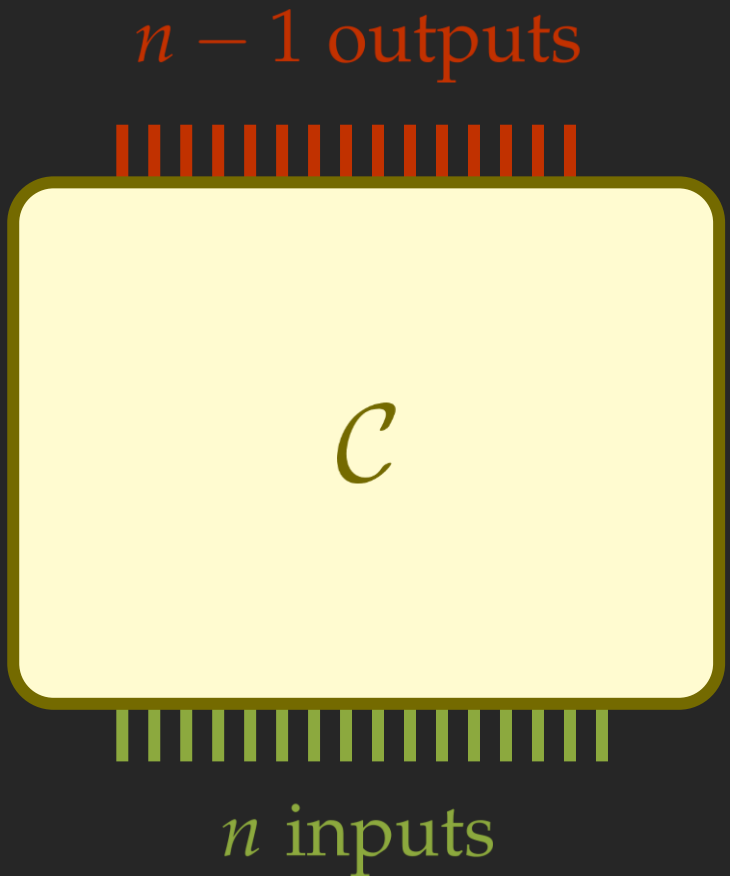
The Inadequacy of NP-hardness

Collision Resistant Hash Functions



The Inadequacy of NP-hardness

Collision Resistant Hash Functions



Hard to find x, x', with $x \neq x'$ and $\mathcal{C}(x) = \mathcal{C}(x')$

The Inadequacy of NP-hardness

To achieve cryptographic utopia for Collision Resistant Hash Functions we have to prove hardness for **search** problems that are **total**!

The Inadequacy of NP-hardness

To achieve cryptographic utopia for Collision Resistant Hash Functions we have to prove hardness for **search** problems that are **total**!

Total Search Problem: the answer to the decision version of the problem is always affirmative, i.e. solution is guaranteed to exist.

The Inadequacy of NP-hardness

To achieve cryptographic utopia for Collision Resistant Hash Functions we have to prove hardness for **search** problems that are **total**!

Total Search Problem: the answer to the decision version of the problem is always affirmative, i.e. solution is guaranteed to exist.

e.g. Any compressing function always has a collision!

The Inadequacy of NP-hardness

To achieve cryptographic utopia for Collision Resistant Hash Functions we have to prove hardness for **search** problems that are **total**!

Theorem **[Johnson Papadimitriou Yannakakis '88, Meggido Papadimitirou '91]**
If a total search problem is NP-hard then NP = co-NP.

The Inadequacy of NP-hardness

To achieve cryptographic utopia for Collision Resistant Hash Functions we have to prove hardness for **search** problems that are **total**!

Theorem [Johnson Papadimitriou Yannakakis '88, Meggido Papadimitirou '91]
If a total search problem is NP-hard then NP = co-NP.

We cannot hope to use NP-hardness!

(

The Inadequacy of NP-hardness

Theorem [Johnson Papadimitriou Yannakakis ‘88, Meggido Papadimitirou ‘91]
If a total search problem is NP-hard then NP = co-NP.

What about **randomized** reductions?

The Inadequacy of NP-hardness

Theorem [Johnson Papadimitriou Yannakakis '88, Meggido Papadimitirou '91]
If a total search problem is NP-hard then NP = co-NP.

What about **randomized** reductions?

If a total seach problem is NP-hard under randomized reductions then

- *we know*: SAT is **checkable**.

The Inadequacy of NP-hardness

Theorem [Johnson Papadimitriou Yannakakis ‘88, Meggido Papadimitirou ‘91]
If a total search problem is NP-hard then NP = co-NP.

What about **randomized** reductions?

If a total seach problem is NP-hard under randomized reductions then

- *we know*: SAT is **checkable**.
- *we want:* AM = co-AM, implies PH collapses [Hastad, Boppana, Zachos ‘87].

The Inadequacy of NP-hardness

Theorem [Johnson Papadimitriou Yannakakis '88, Meggido Papadimitirou '91]
If a total search problem is NP-hard then NP = co-NP.

What about **randomized** reductions?

If a total seach problem is NP-hard under randomized reductions then

- *we know*: SAT is **checkable**.
- *we want:* AM = co-AM, implies PH collapses [Hastad, Boppana, Zachos '87].
  PH collapses directly.

)

The Inadequacy of NP-hardness

To achieve cryptographic utopia for Collision Resistant Hash Functions we have to prove hardness for **search** problems that are **total**!

Theorem [Johnson Papadimitriou Yannakakis '88, Meggido Papadimitirou '91]
If a total search problem is NP-hard then NP = co-NP.

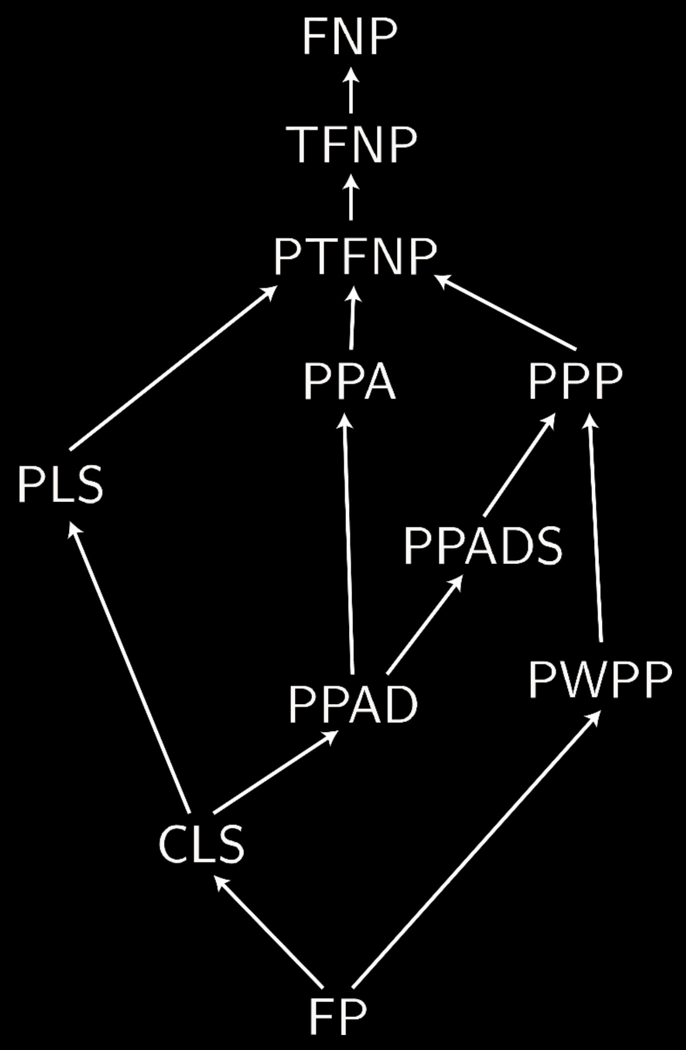
We cannot hope to use NP-hardness!

Complexity of Total Search Problems

FNP: class of search problems whose decision version is in NP.

TFNP: class of total search problems of FNP, i.e. a solution always exists [MP91]

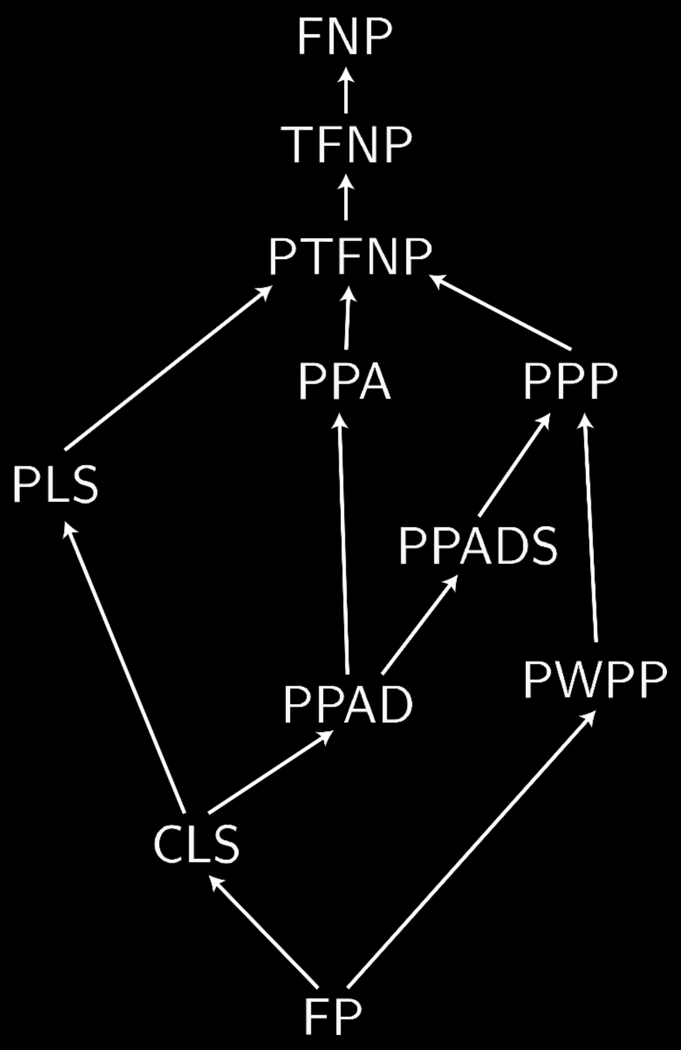
Subclasses of TFNP introduced by [JPY88, **Pap94**, CD11, Jerabek16]



Complexity of Total Search Problems

Many applications in game theory, economics, social choice, (discrete / continuous) optimization,

e.g. [JYP88], [BCE+98], [EGG06], [CDDT09], [DP11], [R15], [R16], [BIQ+17], [GP17], [DTZ18], [FG18] …

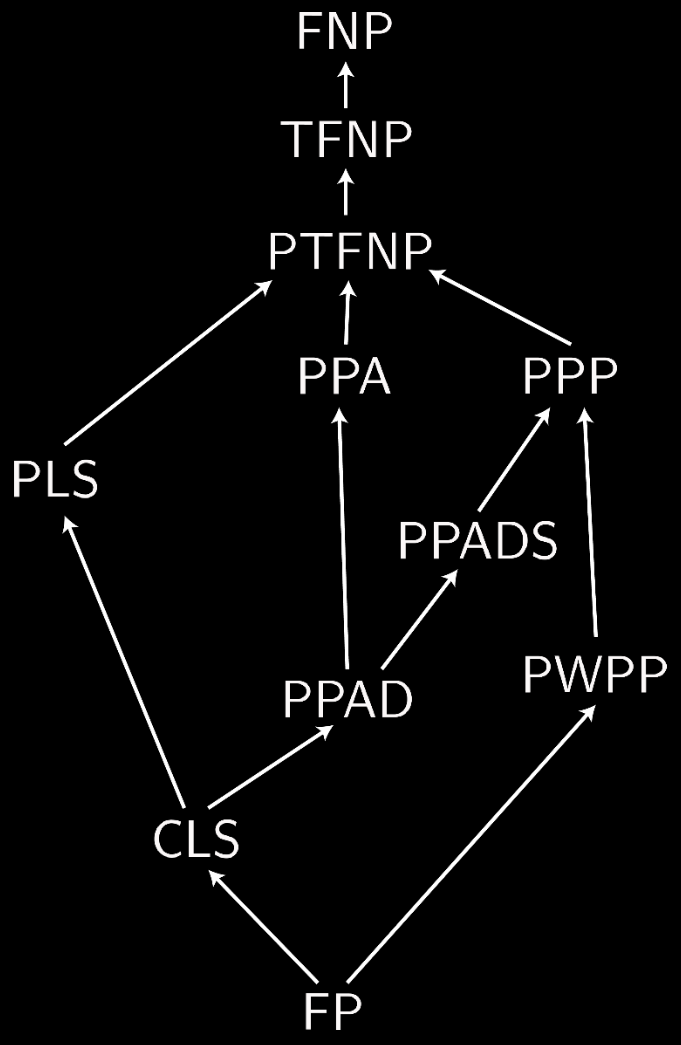


Complexity of Total Search Problems

Many applications in game theory,
economics, social choice,
(discrete / continuous) optimization,
e.g. [JYP88], [BCE+98], [EGG06], [CDDT09], [DP11], [R15], [R16],
[BIQ+17], [GP17], [DTZ18], [FG18] …

Most celebrated result:

 NASH is PPAD-complete [DGP06], [CDT06]



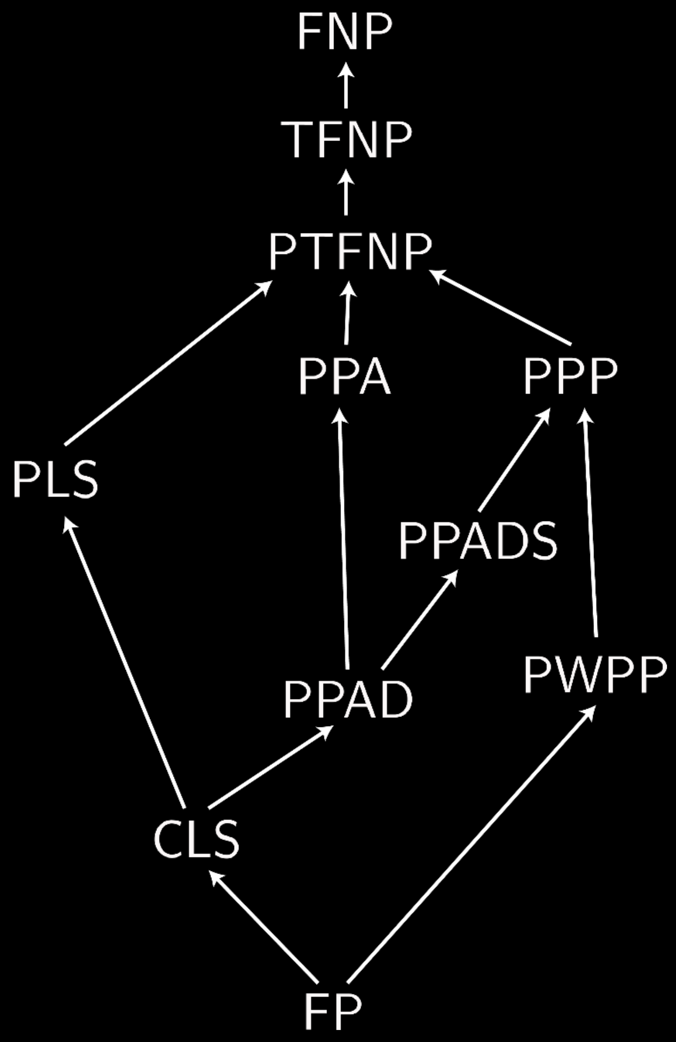
Complexity of Total Search Problems

Many applications in game theory, economics, social choice, (discrete / continuous) optimization,
e.g. [JYP88], [BCE+98], [EGG06], [CDDT09], [DP11], [R15], [R16], [BIQ+17], [GP17], [DTZ18], [FG18] …

Most celebrated result:

 NASH is PPAD-complete [DGP06], [CDT06]

Connections to Cryptography:
[Bur06], [BPR15], [Jer16], [GPS16], [HY17], [RSS17], [HNY17], [KNY17]



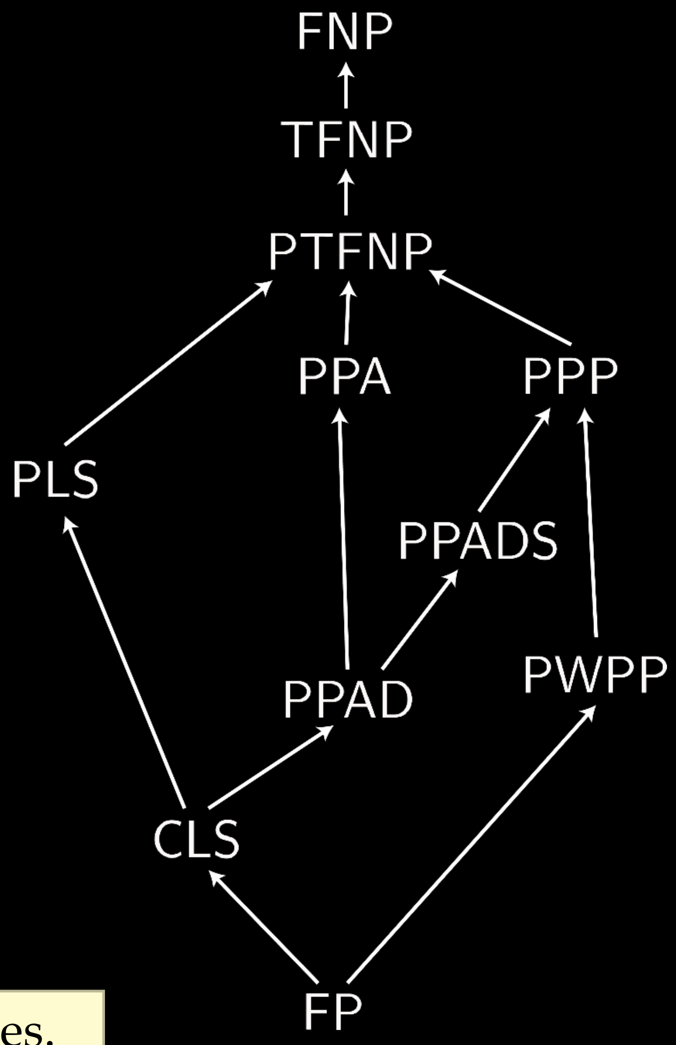
Complexity of Total Search Problems

Many applications in game theory, economics, social choice, (discrete / continuous) optimization,
e.g. [JYP88], [BCE+98], [EGG06], [CDDT09], [DP11], [R15], [R16], [BIQ+17], [GP17], [DTZ18], [FG18] …

Most celebrated result:

 NASH is PPAD-complete [DGP06], [CDT06]

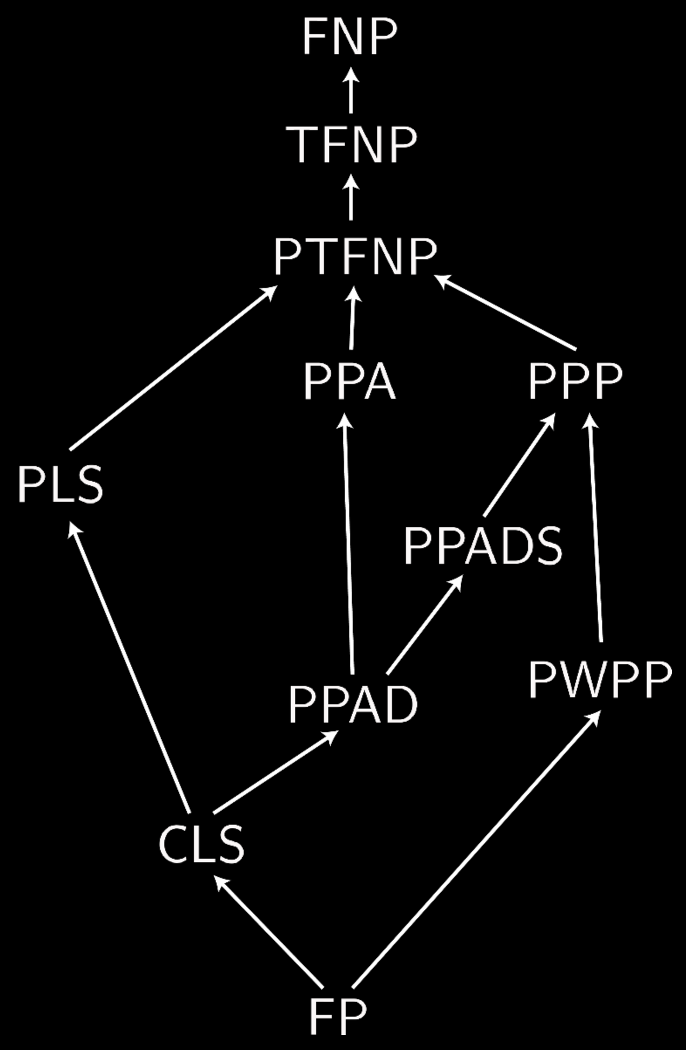
Connections to Cryptography:
[Bur06], [BPR15], [Jer16], [GPS16], [HY17], [RSS17], [HNY17], [KNY17]



You can visit FOCS 2018 workshop on TFNP for references.

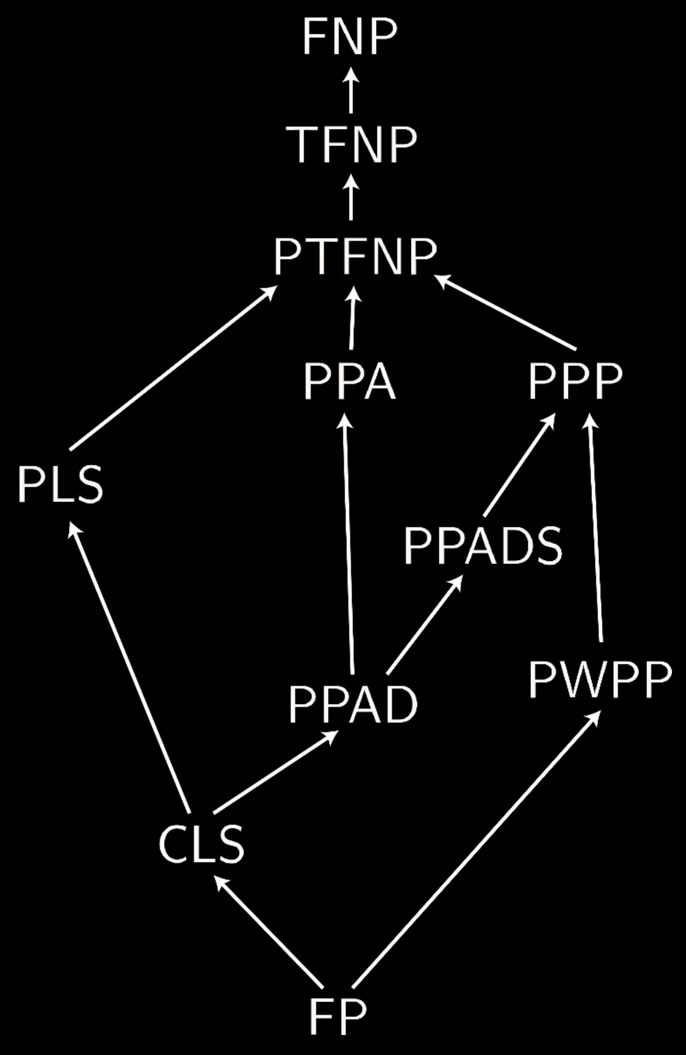
Complexity of Total Search Problems

Prior to our work **natural** complete problems for all subclasses except: PPP, PWPP, CLS, PPADS.



Complexity of Total Search Problems

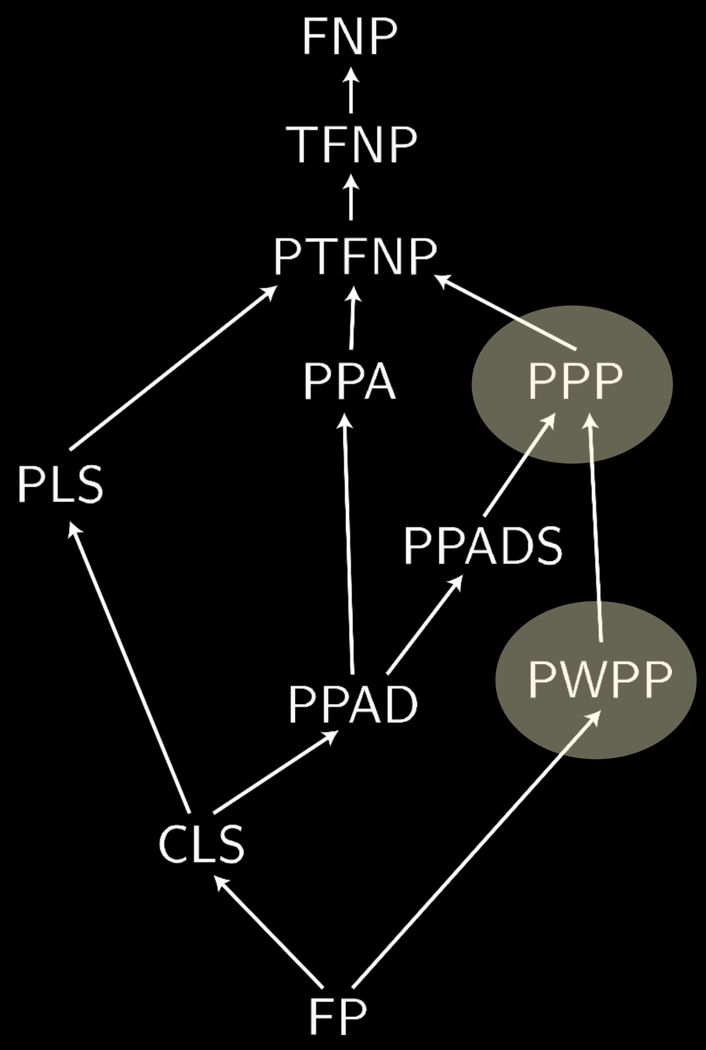
Natural: the problem does not contain a circuit or a Turing machine as part of the input.



Complexity of Total Search Problems

Prior to our work natural complete problems for all subclasses except: PPP, PWPP, CLS, PPADS.

Our Result
We identify the first natural PPP-complete and PWPP-complete problems answering an open problem since [Pap94].



Polynomial Pigeonhole Principle

"Total search problems should be classified in terms of the profound mathematical principles that are invoked to establish their totality."

Papadimitriou '94

PPP, PWPP $\longrightarrow$ Pigeonhole principle

Polynomial Pigeonhole Principle

PPP:

Given a circuit $C : \{0,1\}^n \to \{0,1\}^n$. Find:

Polynomial Pigeonhole Principle

PPP:
Given a circuit $C : \{0,1\}^n \to \{0,1\}^n$. Find:

1. An $\mathbf{x}$ s.t. $C(\mathbf{x}) = \mathbf{0}$

Polynomial Pigeonhole Principle

PPP:

Given a circuit $C : \{0,1\}^n \to \{0,1\}^n$. Find:

1. An $\mathbf{x}$ s.t. $C(\mathbf{x}) = \mathbf{0}$ or
2. a collision, i.e $\mathbf{x} \neq \mathbf{y}$ s.t. $C(\mathbf{x}) = C(\mathbf{y})$.

Polynomial Pigeonhole Principle

PPP:
Given a circuit $C : \{0,1\}^n \to \{0,1\}^n$. Find:

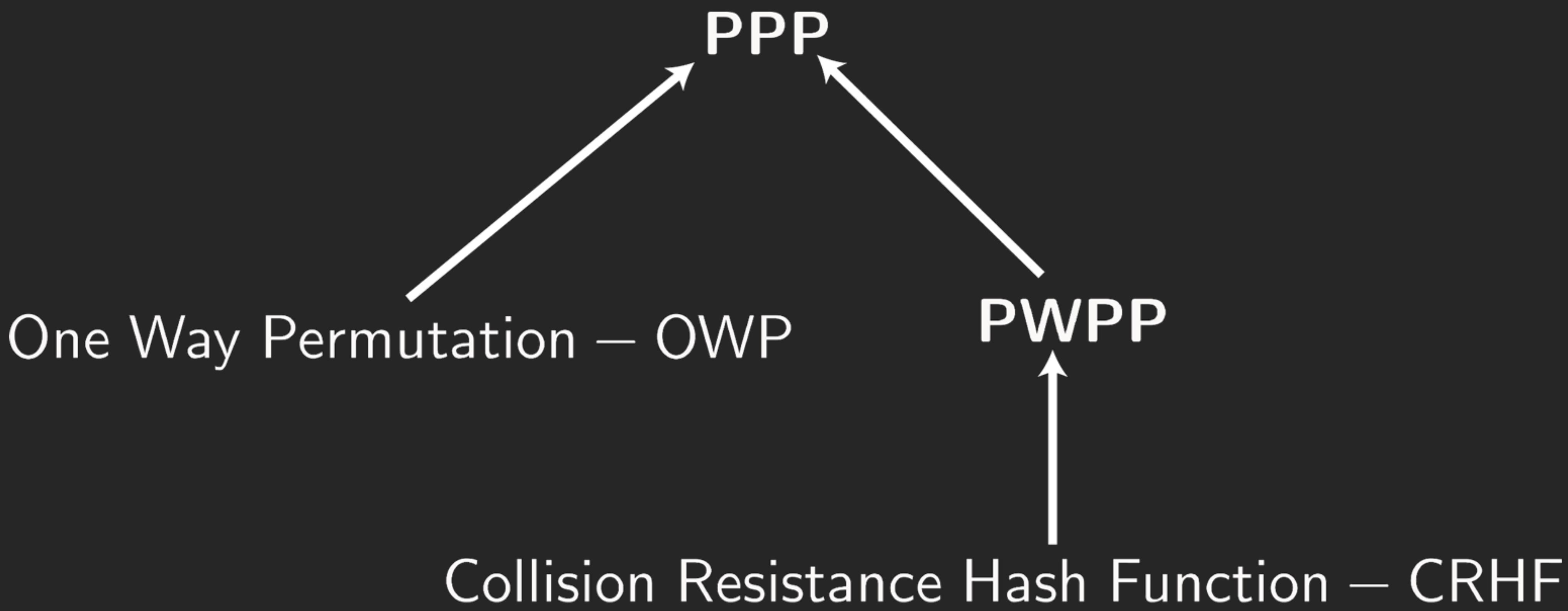
1. An $\mathbf{x}$ s.t. $\mathsf{C}(\mathbf{x}) = \mathbf{0}$ or
2. a collision, i.e $\mathbf{x} \neq \mathbf{y}$ s.t. $\mathsf{C}(\mathbf{x}) = \mathsf{C}(\mathbf{y})$.

Obviously a total problem, cannot be NP-hard!

Polynomial Pigeonhole Principle

PWPP:
Given a circuit $C : \{0,1\}^n \to \{0,1\}^m$, with $m < n$.
Find a collision, i.e $\mathbf{x} \neq \mathbf{y}$ s.t. $C(\mathbf{x}) = C(\mathbf{y})$.

PPP/PWPP and Cryptography



PPP/PWPP-completeness

A longstanding open problem since [Papadimitriou '94].

Our contribution:
We identify the first natural PPP/PWPP-complete problems.

This talk: **PWPP**.

Main Theorem:
WEAK-cSIS is PWPP-complete.

Short Integer Solution (SIS) Problem

INPUT: $\boxed{\mathbf{A}} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}$ s.t. $\|\mathbf{x}\| \leq \beta$, $\mathbf{A}\mathbf{x} = \mathbf{0} \pmod{q}$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}$ s.t. $\|\mathbf{x}\| \leq \beta$, $\mathbf{A}\mathbf{x} = \mathbf{0} \pmod{q}$

$\boxed{\mathbf{x} \neq \mathbf{0}}$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}$ s.t. $\|\mathbf{x}\| \leq \beta$, $\mathbf{A}\mathbf{x} = \mathbf{0} \pmod q$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}$ s.t. $\|\mathbf{x}\| \leq 1$, $\mathbf{A}\mathbf{x} = \mathbf{0} \pmod q$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod{q}$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod{q}$

domain size is 2^m

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod{q}$

image size is q^r

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $2^m > q^r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

Short Integer Solution (SIS) Problem

The problem is total!

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $2^m > q^r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod{q}$

Short Integer Solution (SIS) Problem

The problem is in PWPP!

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $2^m > q^r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

Short Integer Solution (SIS) Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)r$.

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\,\mathbf{x} = \mathbf{A}\,\mathbf{y} \pmod q$

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)(r + d)$ $\mathbf{G} \in \mathbb{Z}_q^{d \times m}$, and *binary invertible*

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d \times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)(r+d)$ $\mathbf{G} \in \mathbb{Z}_q^{d \times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

Why is this problem total?

Constraint Short Integer Solution Problem

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

Why is this problem total?

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

Binary Invertible Matrix

Binary Invertible Matrix

Binary Invertible Matrix

$$\mathbf{G} = \begin{bmatrix} \mathbf{g} & & & \star & \star & \star \\ & \mathbf{g} & & & & \\ \mathbf{0} & & \mathbf{g} & & & \end{bmatrix}$$

$$\mathbf{g} = \begin{bmatrix} 1 & 2 & 4 & \dots & 2^{k-1} \end{bmatrix} \qquad 2^k \geq q$$

Binary Invertible Matrix

$$\mathbf{g} = \begin{bmatrix} 1 & 2 & 4 & \dots & 2^{k-1} \end{bmatrix} \qquad 2^k \geq q$$

e.g. for m = 10, q = 8

$$\mathbf{G} = \begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix}$$

Binary Invertible Matrix

$$\mathbf{G} = \begin{bmatrix} \mathbf{g} & & & \star & \star & \star \\ & \mathbf{g} & & & & \\ \mathbf{0} & & \mathbf{g} & & & \end{bmatrix}$$

$$\mathbf{g} = \begin{bmatrix} 1 & 2 & 4 & \ldots & 2^{k-1} \end{bmatrix} \qquad 2^k \geq q$$

e.g. for m = 10, q = 8

$$\mathbf{G} = \begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix}$$

Binary Invertible Matrix

Lemma
For any $\mathbf{z} \in \{0,1\}^{m-\log(q)d}$ and any $\mathbf{b} \in \mathbb{Z}_q^d$, we can efficiently compute a binary solution of the form $\mathbf{x} = [\star \;\; \star \cdots \star \;\; \mathbf{z}]$ for the equation $\mathbf{G}\mathbf{x} = \mathbf{b} \pmod q$.

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ z \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod{8}$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & \boxed{1} & \boxed{2} & \boxed{4} & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \boxed{x_7} \\ \boxed{x_8} \\ \boxed{x_9} \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & \boxed{1 \quad 2 \quad 4} & & & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ x_7 \\ x_8 \\ x_9 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

$$1 \cdot x_7 + 2 \cdot x_8 + 4 \cdot x_9 = 1 \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & \boxed{1 \quad 2 \quad 4} & & & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ \boxed{1} \\ \boxed{0} \\ \boxed{0} \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \star \\ \star \\ \star \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ x_4 \\ x_5 \\ x_6 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & \boxed{1 \quad 2 \quad 4} & & & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ x_4 \\ x_5 \\ x_6 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & \boxed{1 \quad 2 \quad 4} & & & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ \boxed{\begin{matrix} 1 \\ 0 \\ 0 \end{matrix}} \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod{8}$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} \star \\ \star \\ \star \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

$$\begin{bmatrix} \boxed{1 & 2 & 4} & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

Example

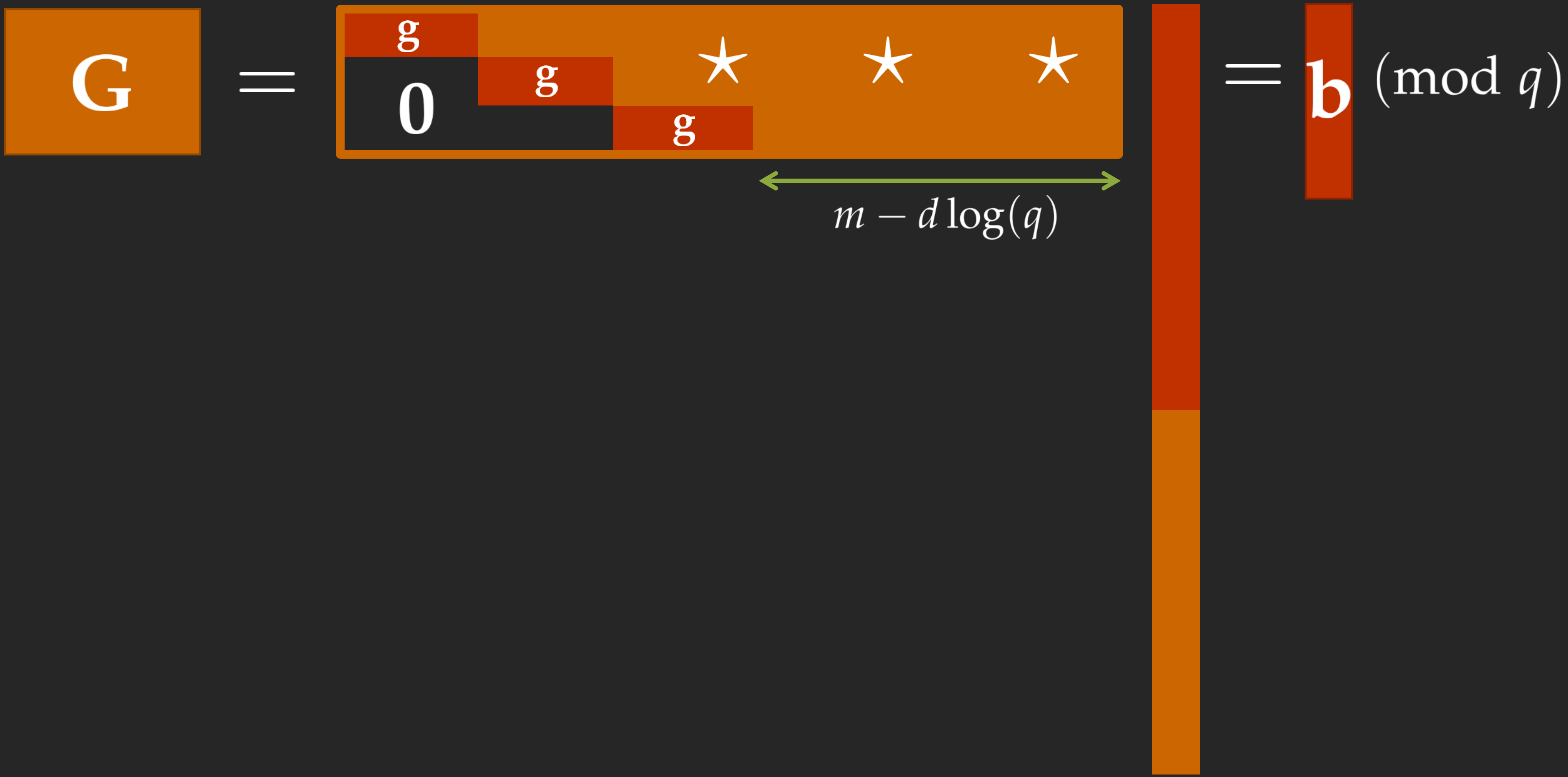
$$\begin{bmatrix} \boxed{1 & 2 & 4} & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

Binary Invertible Matrix

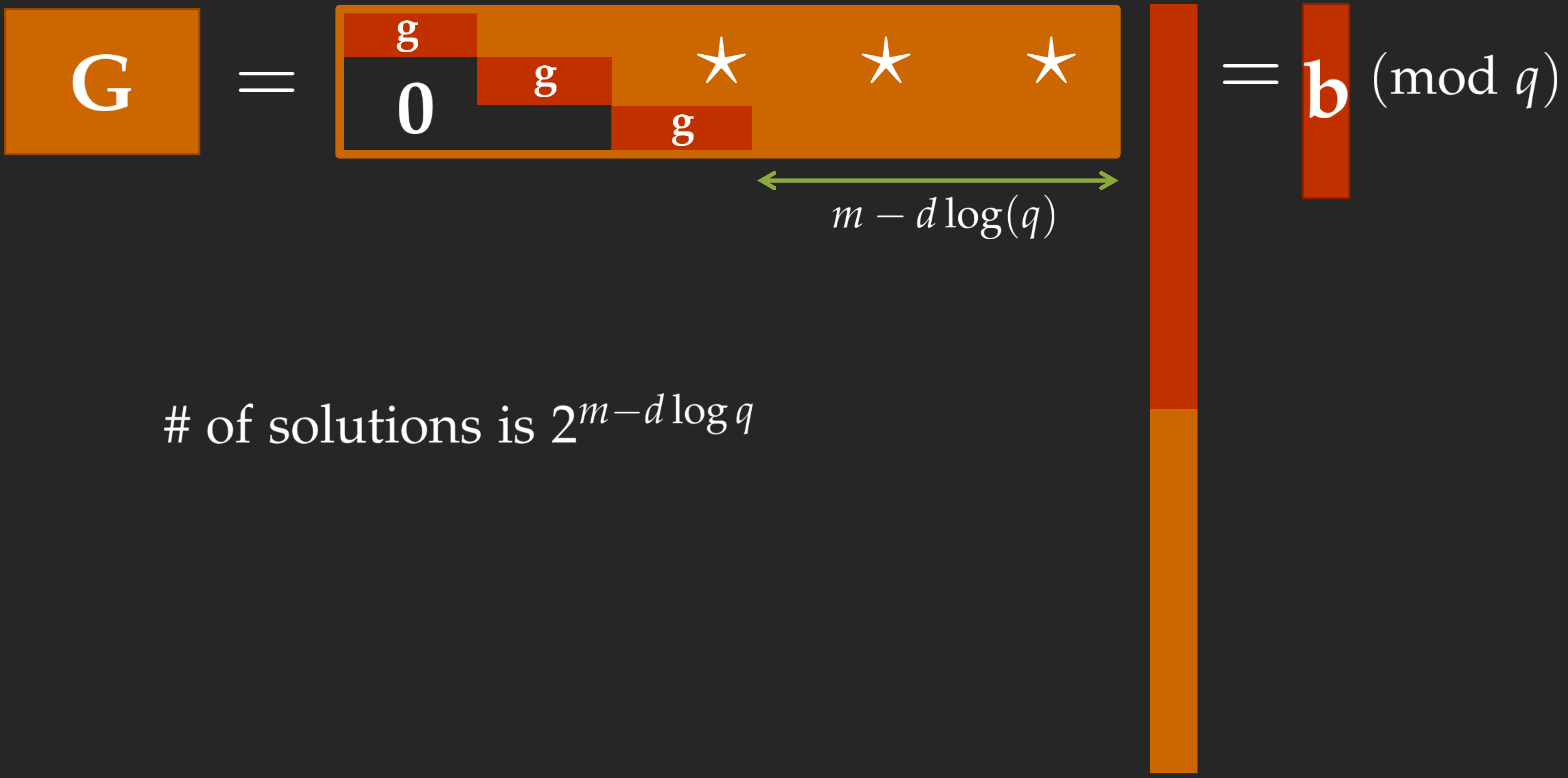
Example

$$\begin{bmatrix} 1 & 2 & 4 & 3 & 0 & 6 & 5 & 6 & 2 & 1 \\ 0 & 0 & 0 & 1 & 2 & 4 & 1 & 0 & 3 & 5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 4 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \\ 1 \end{bmatrix} \pmod 8$$

WEAK-cSIS is Total



WEAK-cSIS is Total



# of solutions is $2^{m-d\log q}$

WEAK-cSIS is Total

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

WEAK-cSIS is Total

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$

WEAK-cSIS is Total

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m - d\log(q) > r\log(q)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d \times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$

WEAK-cSIS is Total

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m - d\log(q) > r\log(q)$ $\quad$ $\mathbf{G} \in \mathbb{Z}_q^{d \times m}$, and *binary invertible*

OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$$

$\blacksquare$

WEAK-CSIS is in PWPP

Lemma
For any $\mathbf{z} \in \{0,1\}^{m-\log(q)d}$ and any $\mathbf{b} \in \mathbb{Z}_q^d$, we can efficiently compute a binary solution of the form $\mathbf{x} = [\star \;\; \star \cdots \star \;\; \mathbf{z}]$ for the equation $\mathbf{G}\mathbf{x} = \mathbf{b} \pmod q$.

WEAK-cSIS is in PWPP

Lemma
For any $\mathbf{z} \in \{0,1\}^{m-\log(q)d}$ and any $\mathbf{b} \in \mathbb{Z}_q^d$, we can efficiently compute a binary solution of the form $\mathbf{x} = [\star \ \ \star \cdots \star \ \ \mathbf{z}]$ for the equation $\mathbf{Gx} = \mathbf{b} \pmod q$.

Since $m > (r+d)\log(q)$, there exist more than $2^{\log(q)r} = q^r$, $\mathbf{x} \in \{0,1\}^m$ such that $\mathbf{Gx} = \mathbf{b} \pmod q$.

WEAK-cSIS is in PWPP

Lemma
For any $\mathbf{z} \in \{0,1\}^{m-\log(q)d}$ and any $\mathbf{b} \in \mathbb{Z}_q^d$, we can efficiently compute a binary solution of the form $\mathbf{x} = [\star \;\; \star \cdots \star \;\; \mathbf{z}]$ for the equation $\mathbf{G}\mathbf{x} = \mathbf{b} \pmod q$.

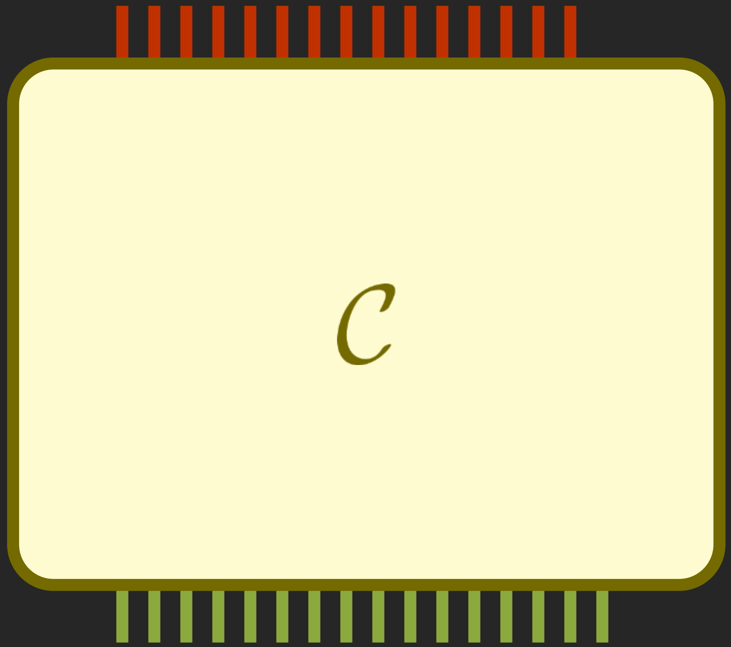
Since $m > (r+d)\log(q)$, there exist more than $2^{\log(q)r} = q^r$, $\mathbf{x} \in \{0,1\}^m$ such that $\mathbf{G}\mathbf{x} = \mathbf{b} \pmod q$.

There exist $\mathbf{x} \neq \mathbf{y}$ such that $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$ and $\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{b} \pmod q$.

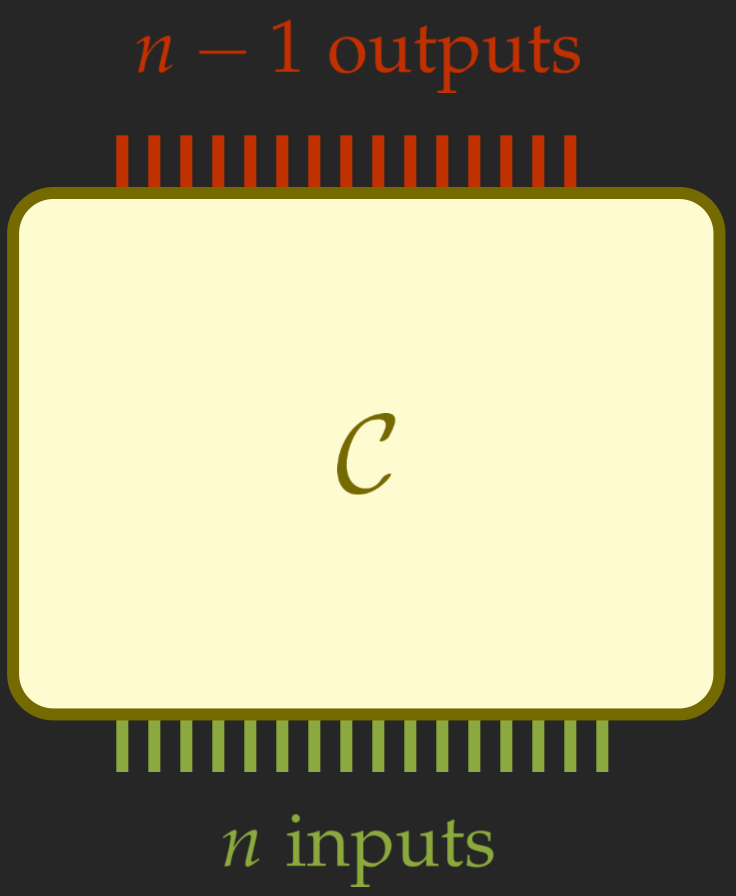
WEAK-CSIS is PWPP-hard

PWPP:
Given a circuit $C : \{0,1\}^n \to \{0,1\}^m$, with $m < n$.
Find a collision, i.e $\mathbf{x} \neq \mathbf{y}$ s.t. $C(\mathbf{x}) = C(\mathbf{y})$.

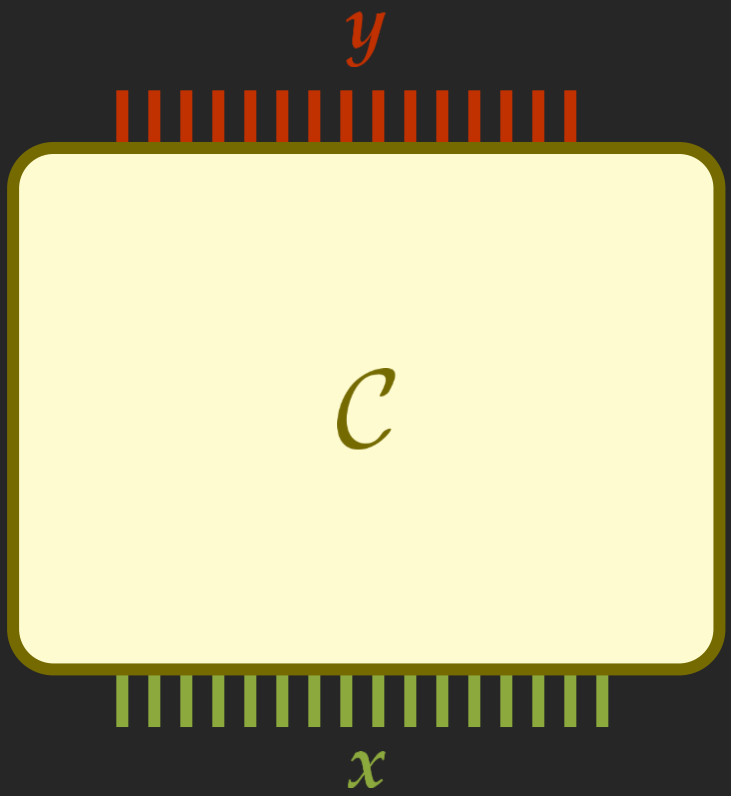
WEAK-CSIS is PWPP-hard



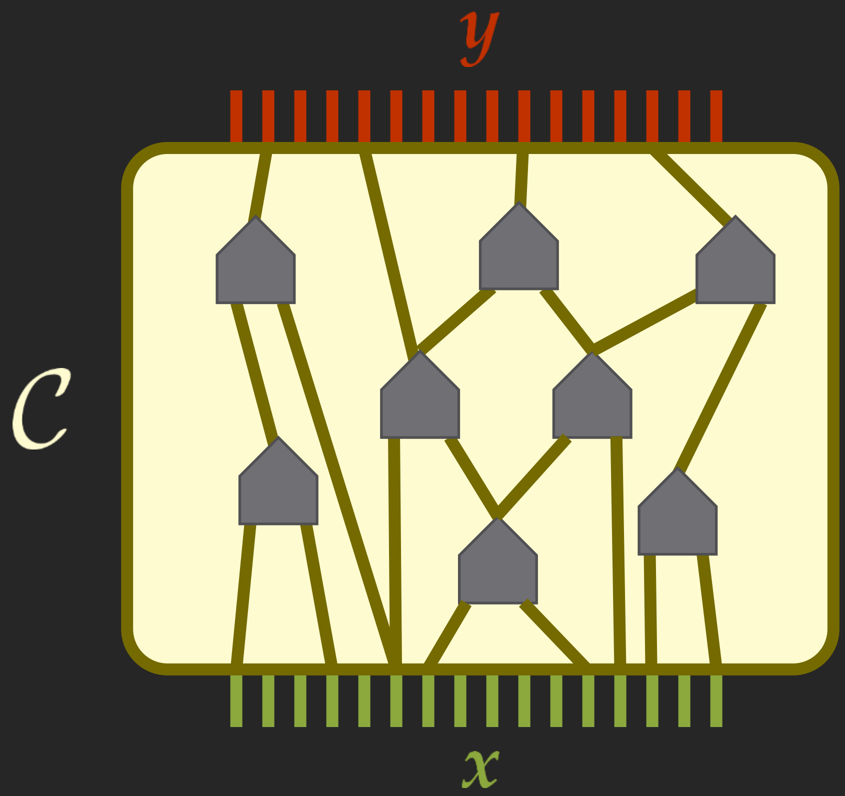
WEAK-cSIS is PWPP-hard



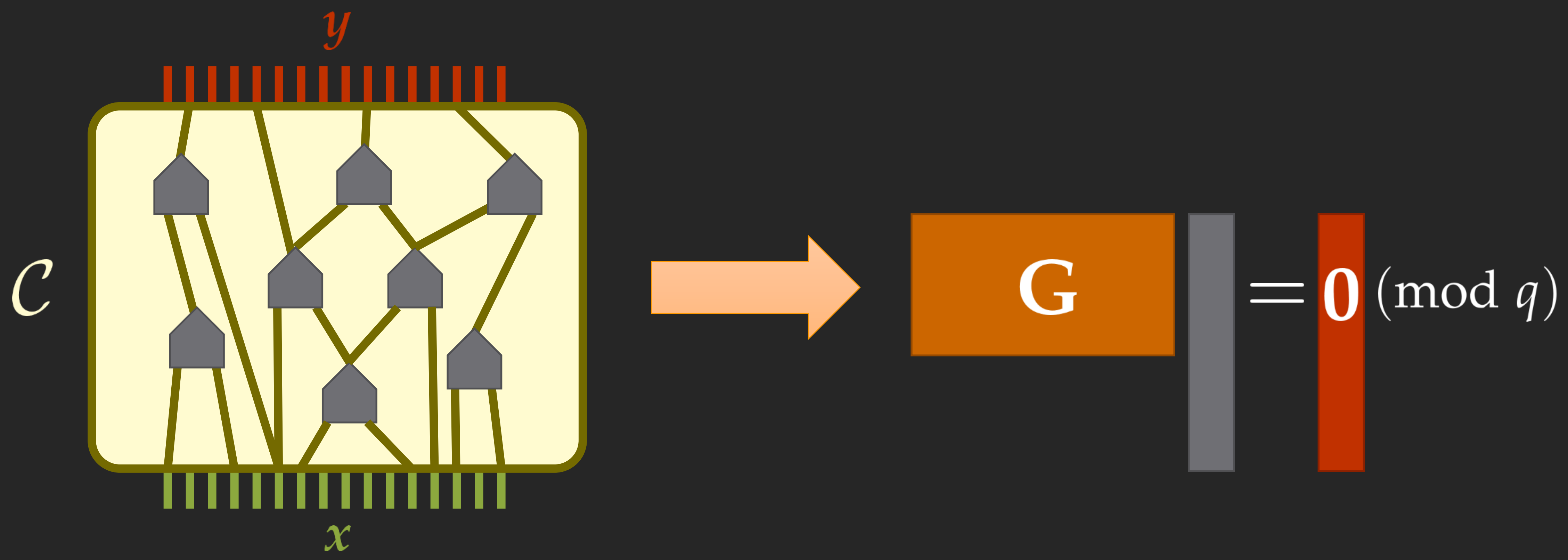
WEAK-cSIS is PWPP-hard



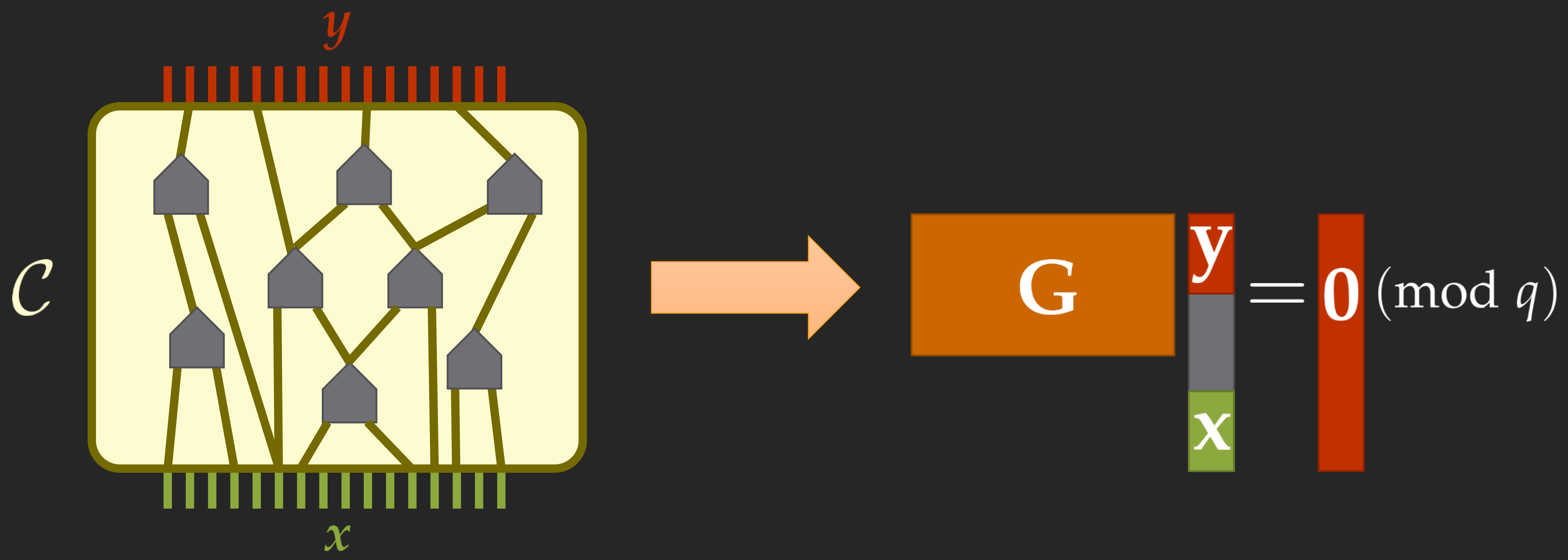
WEAK-cSIS is PWPP-hard



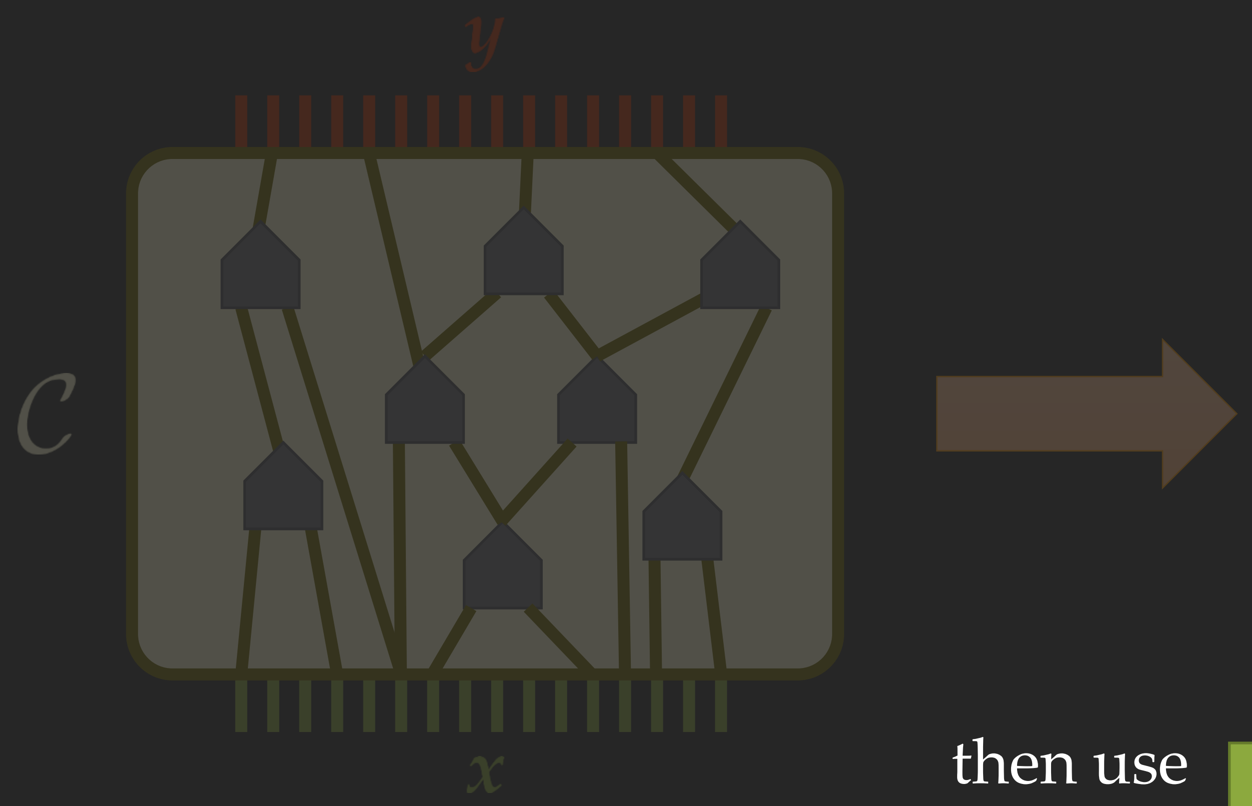
WEAK-cSIS is PWPP-hard

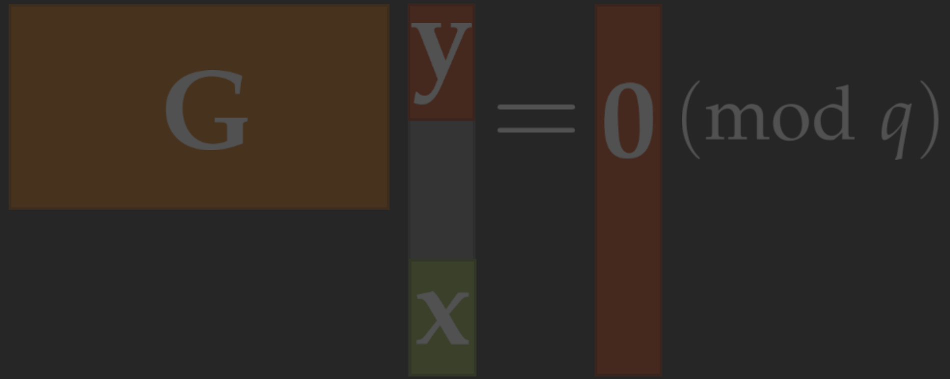


WEAK-cSIS is PWPP-hard



WEAK-cSIS is PWPP-hard

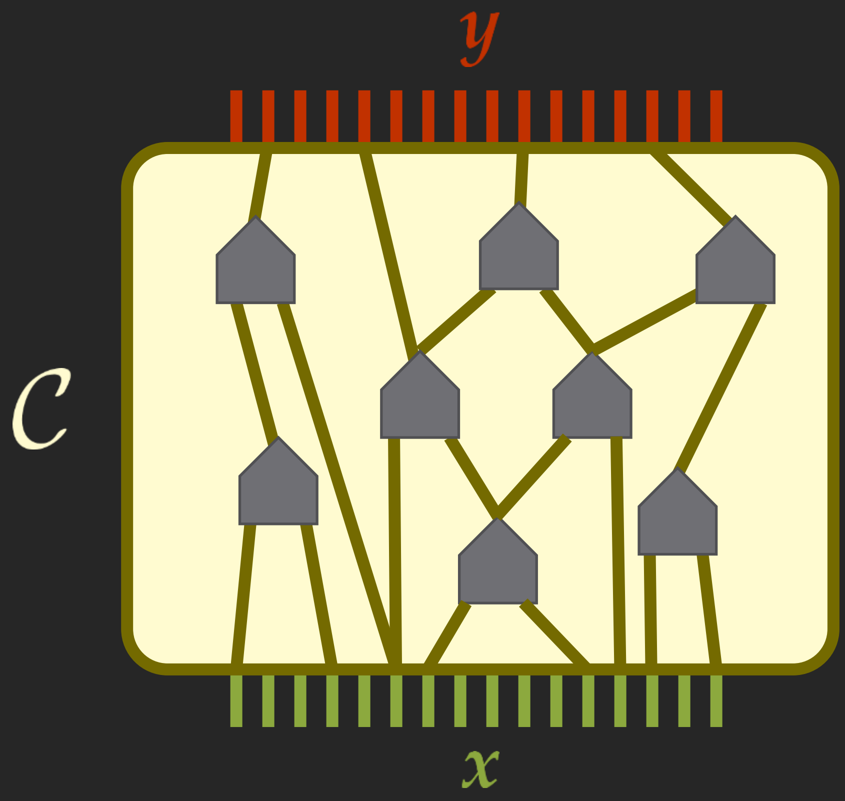




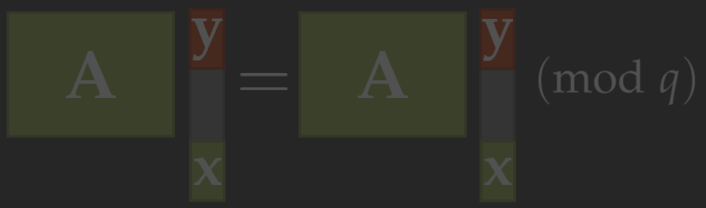
then use $\mathbf{A} \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{y} \\ \mathbf{x} \end{bmatrix} \pmod q$

to find a collision!

WEAK-cSIS is PWPP-hard



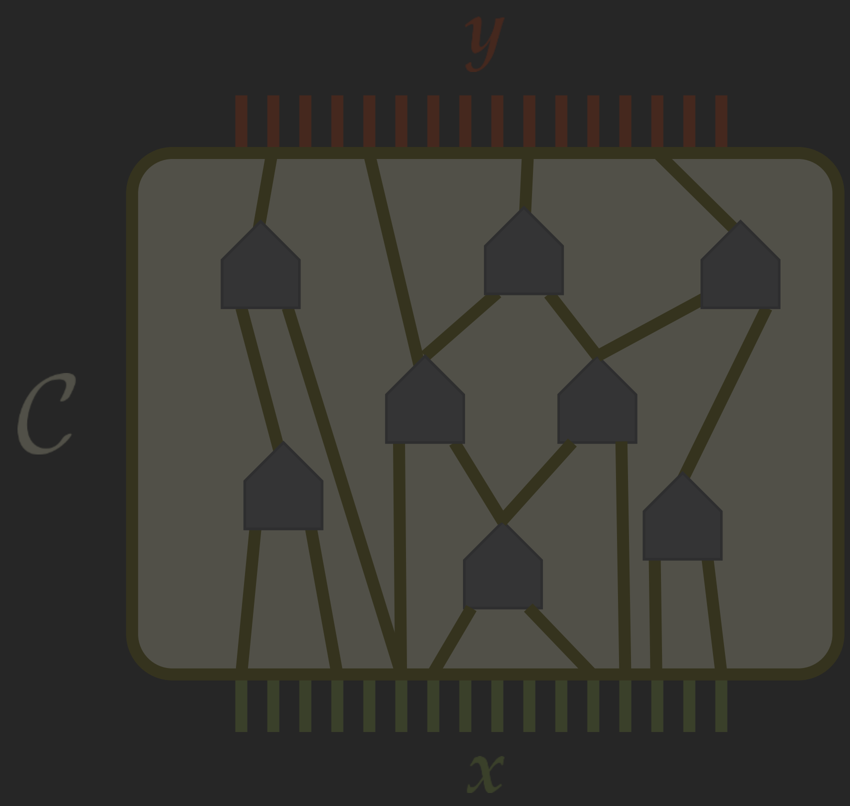
then use



to find a collision!

WEAK-cSIS is PWPP-hard

Attention!
During the reduction we have to preserve **totality**!



$$\mathbf{G}\begin{bmatrix}\mathbf{y}\\ \mathbf{x}\end{bmatrix} = \mathbf{0} \pmod{q}$$

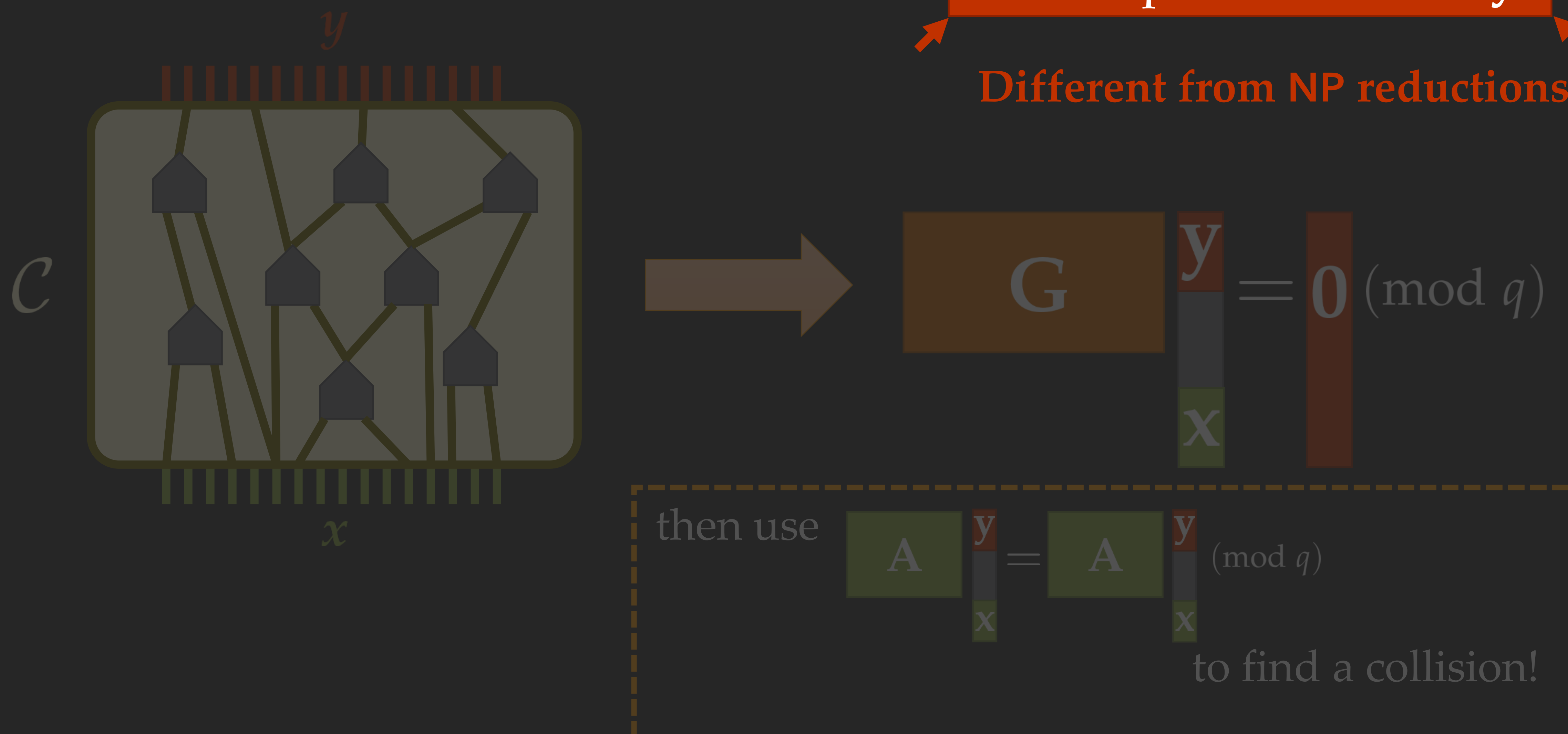
then use $\mathbf{A}\begin{bmatrix}\mathbf{y}\\ \mathbf{x}\end{bmatrix} = \mathbf{A}\begin{bmatrix}\mathbf{y}\\ \mathbf{x}\end{bmatrix} \pmod{q}$

to find a collision!

WEAK-cSIS is PWPP-hard

Attention!
During the reduction we have to preserve **totality**!

Different from NP reductions!



Hash Function from WEAK-CSIS

<u>Hash function family:</u>

- Key: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)(r+d)$ $\mathbf{G} \in \mathbb{Z}_q^{d \times m}$ binary invertible matrix

Hash Function from WEAK-CSIS

<u>Hash function family:</u>

- Key: $\mathbf{A} \in \mathbb{Z}_q^{r \times m}$, with $m > \log(q)(r+d)$ $\quad \mathbf{G} \in \mathbb{Z}_q^{d \times m}$ binary invertible matrix
- Hash(x): For $\mathbf{x} \in \{0,1\}^{m - d\log(q)}$, use Lemma to find $\mathbf{z} \in \{0,1\}^{d\log(q)}$ s.t. $\mathbf{G}\begin{bmatrix}\mathbf{z}\\\mathbf{x}\end{bmatrix} = \mathbf{0} \pmod q$.

$$\mathbf{A}\begin{bmatrix}\mathbf{z}\\\mathbf{x}\end{bmatrix} \pmod q$$

Can we achieve Cryptographic Utopia?

Bottlenecks

- cryptography is based on problems that are hard on average!
- NP-hard problems do not suffice for cryptography.

Can we achieve Cryptographic Utopia?

Bottlenecks

- **cryptography is based on problems that are hard on average!**
- NP-hard problems do not suffice for cryptography.

Approximate Short Integer Solution (APPROXSIS)

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)r$.

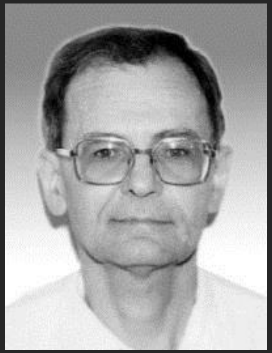
OUTPUT: $\mathbf{x}$ s.t. $\|\mathbf{x}\|_2 \leq B$, $\mathbf{A}\mathbf{x} = \mathbf{0} \pmod{q}$

Average Short Integer Solution (AverageSIS)

INPUT: $\boxed{\mathbf{A}} \sim U\left[\mathbb{Z}_q^{r\times m}\right]$, with $m > \log(q)r$.

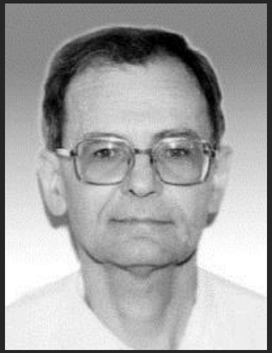
OUTPUT: $\mathbf{x}$ s.t. $\|\mathbf{x}\|_\infty \leq 1$, $\mathbf{A}\mathbf{x} = \mathbf{0} \pmod{q}$

Worst-to-Average Case Reduction for SIS



Informal Theorem[Ajtai'96]
There is a randomized Cook reduction from the **worst-case** problem APPROXSIS to the **average-case** problem AVERAGESIS!

Worst-to-Average Case Reduction for SIS



Informal Theorem [Ajtai'96]
There is a randomized Cook reduction from the **worst-case** problem APPROXSIS to the **average-case** problem AVERAGESIS!

This result is the foundation of lattice based cryptography.

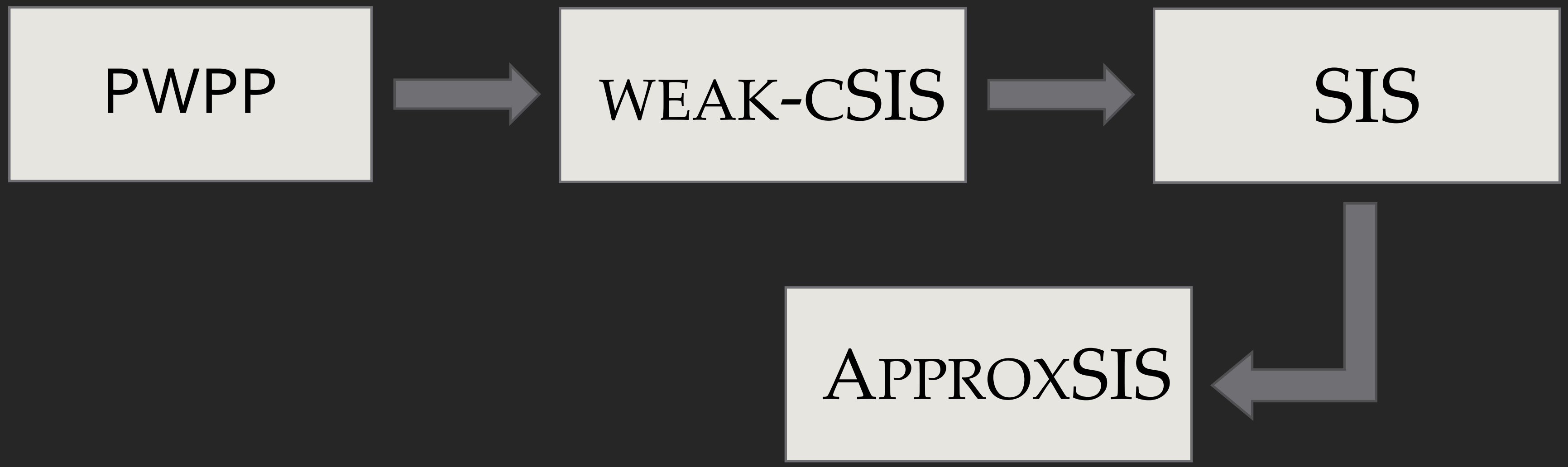
Can we achieve Cryptographic Utopia?

PWPP $\rightarrow$ WEAK-cSIS

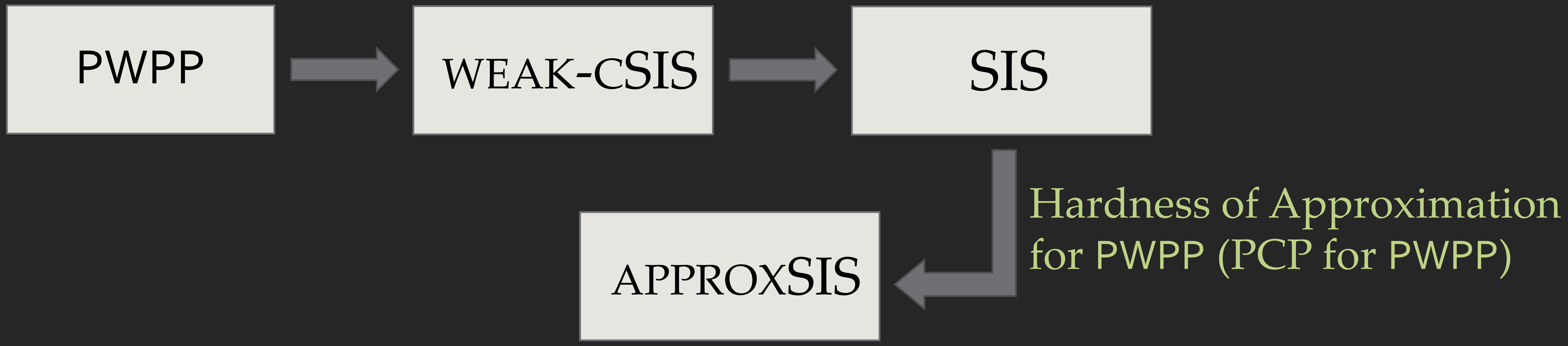
Can we achieve Cryptographic Utopia?

PWPP → WEAK-cSIS → SIS

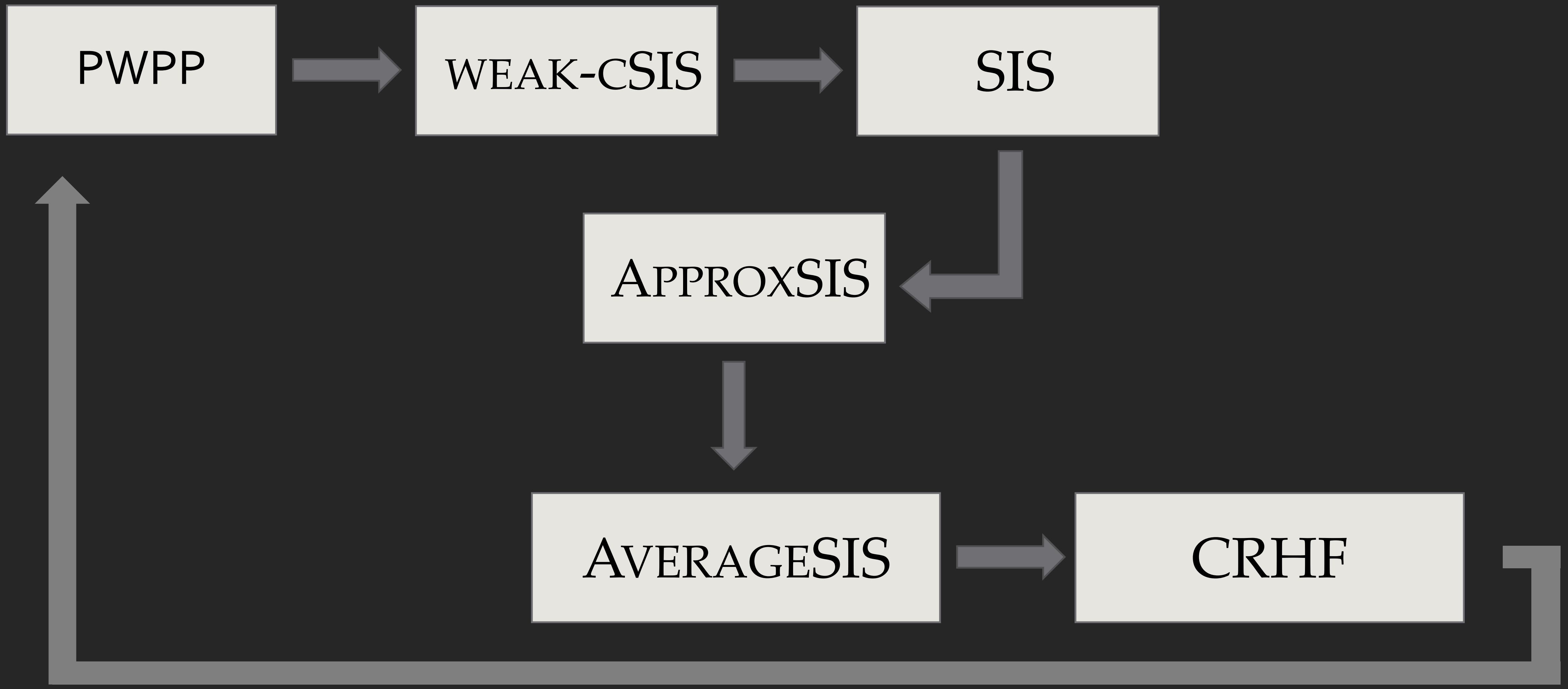
Can we achieve Cryptographic Utopia?



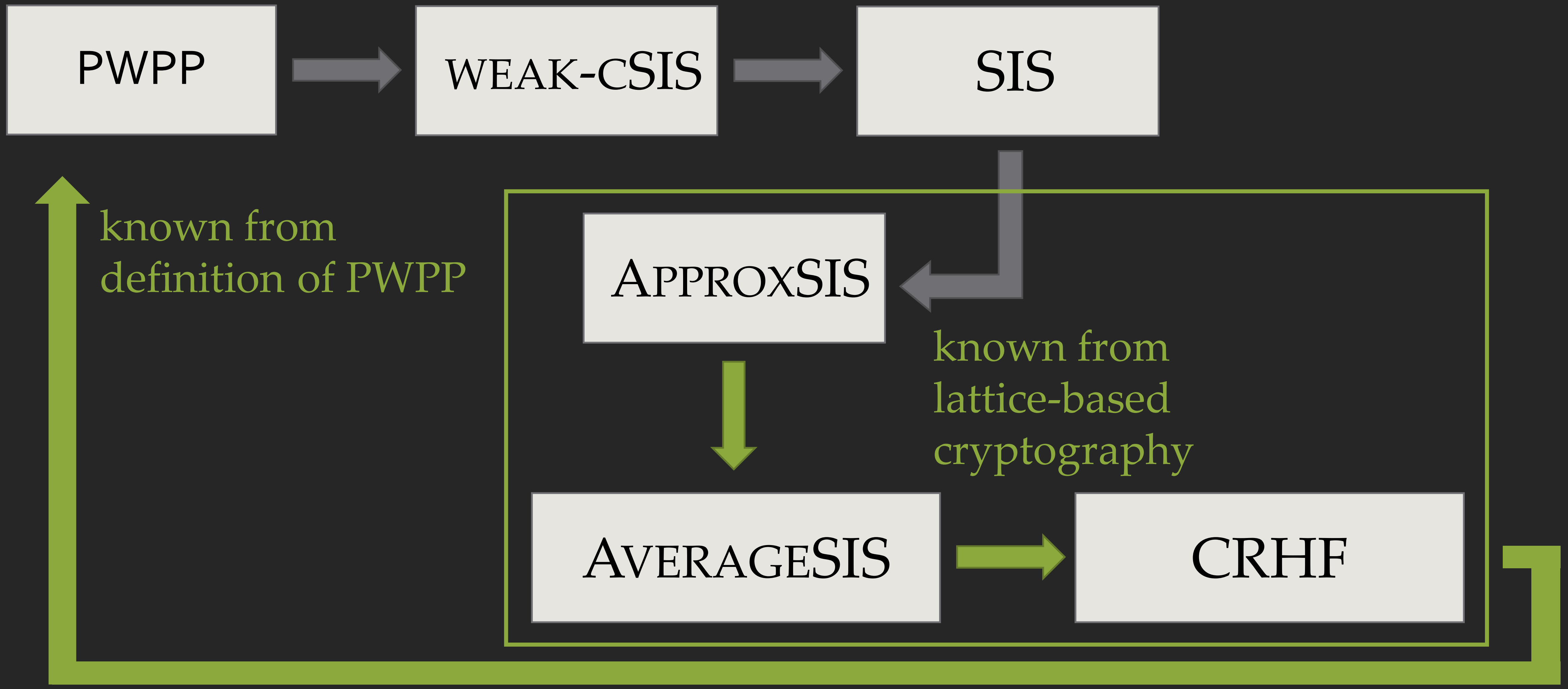
Can we achieve Cryptographic Utopia?



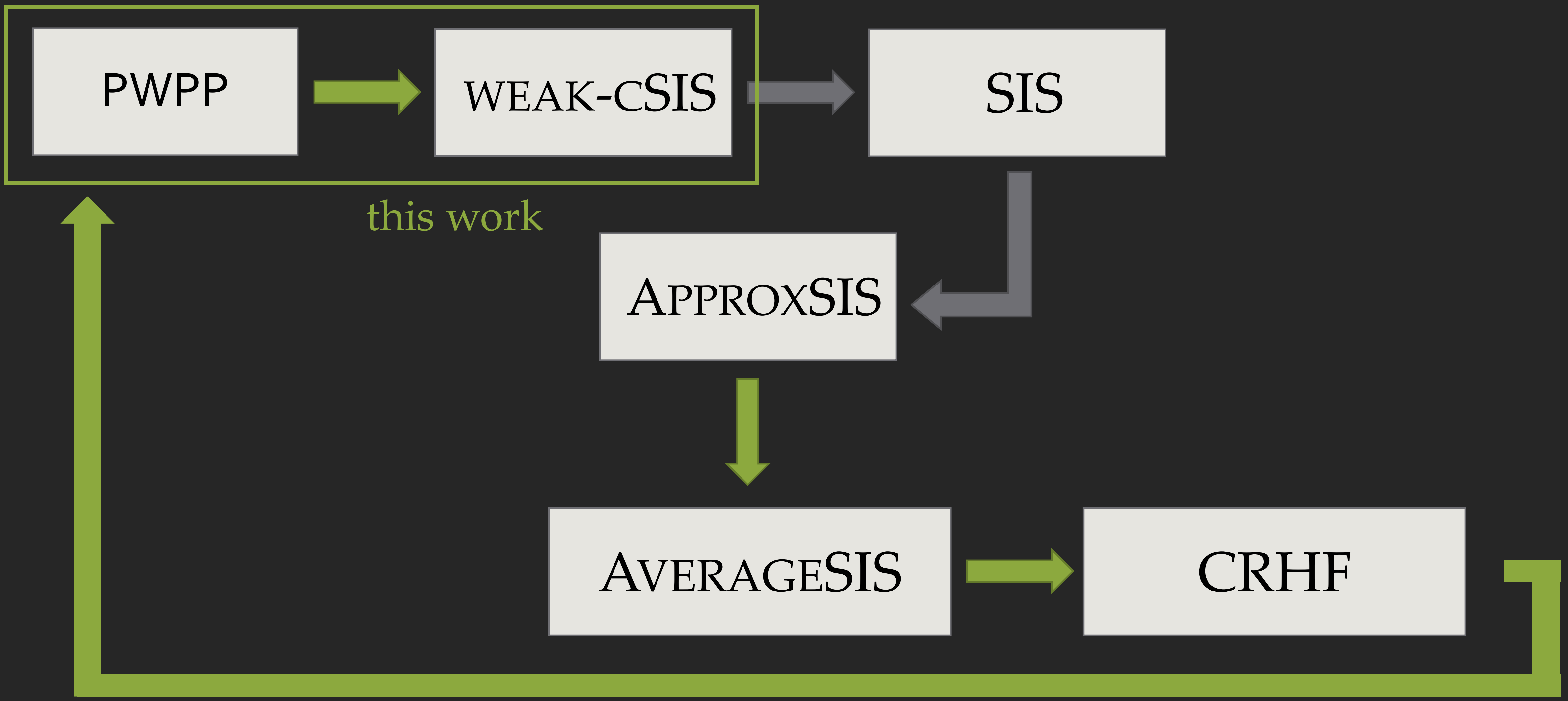
Can we achieve Cryptographic Utopia?



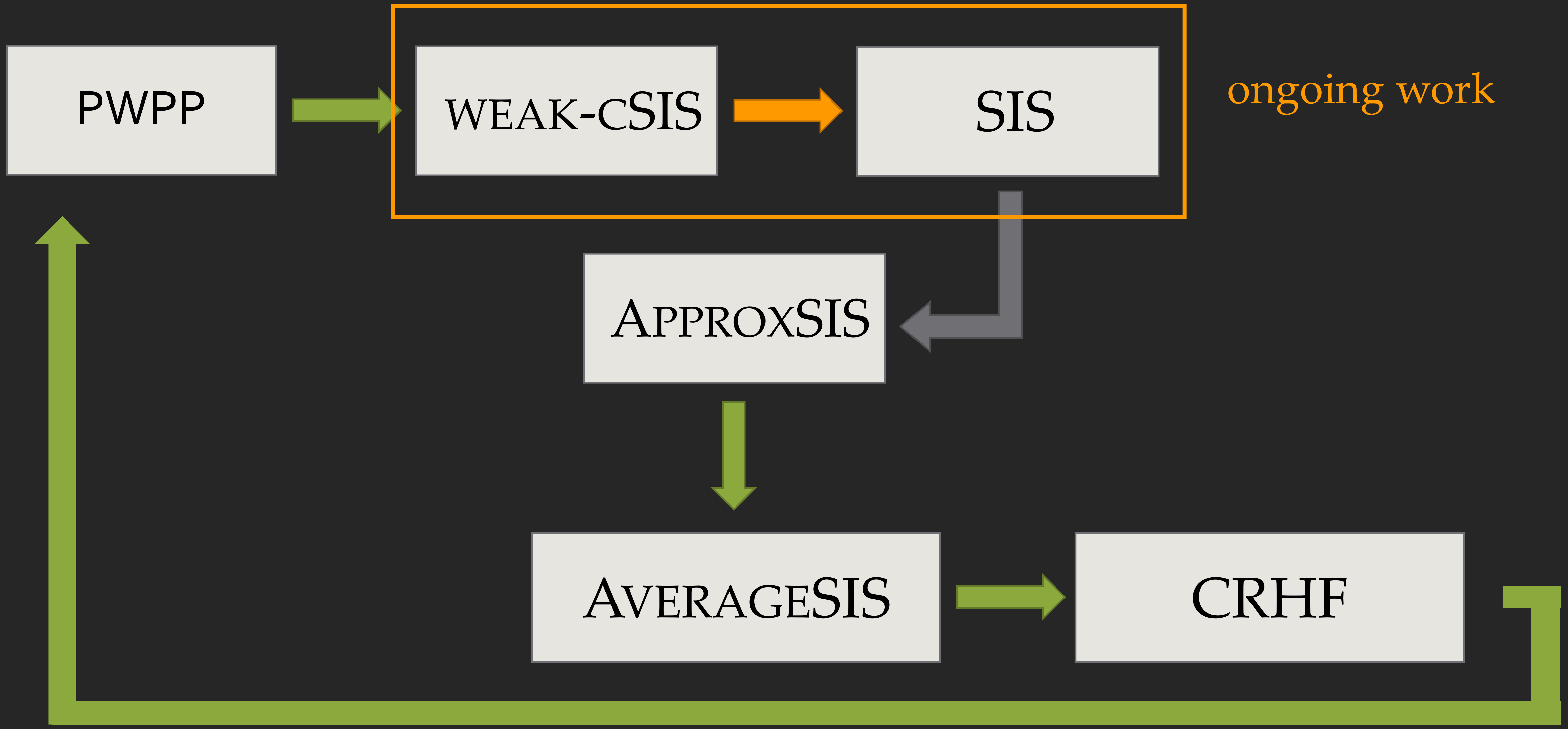
Can we achieve Cryptographic Utopia?



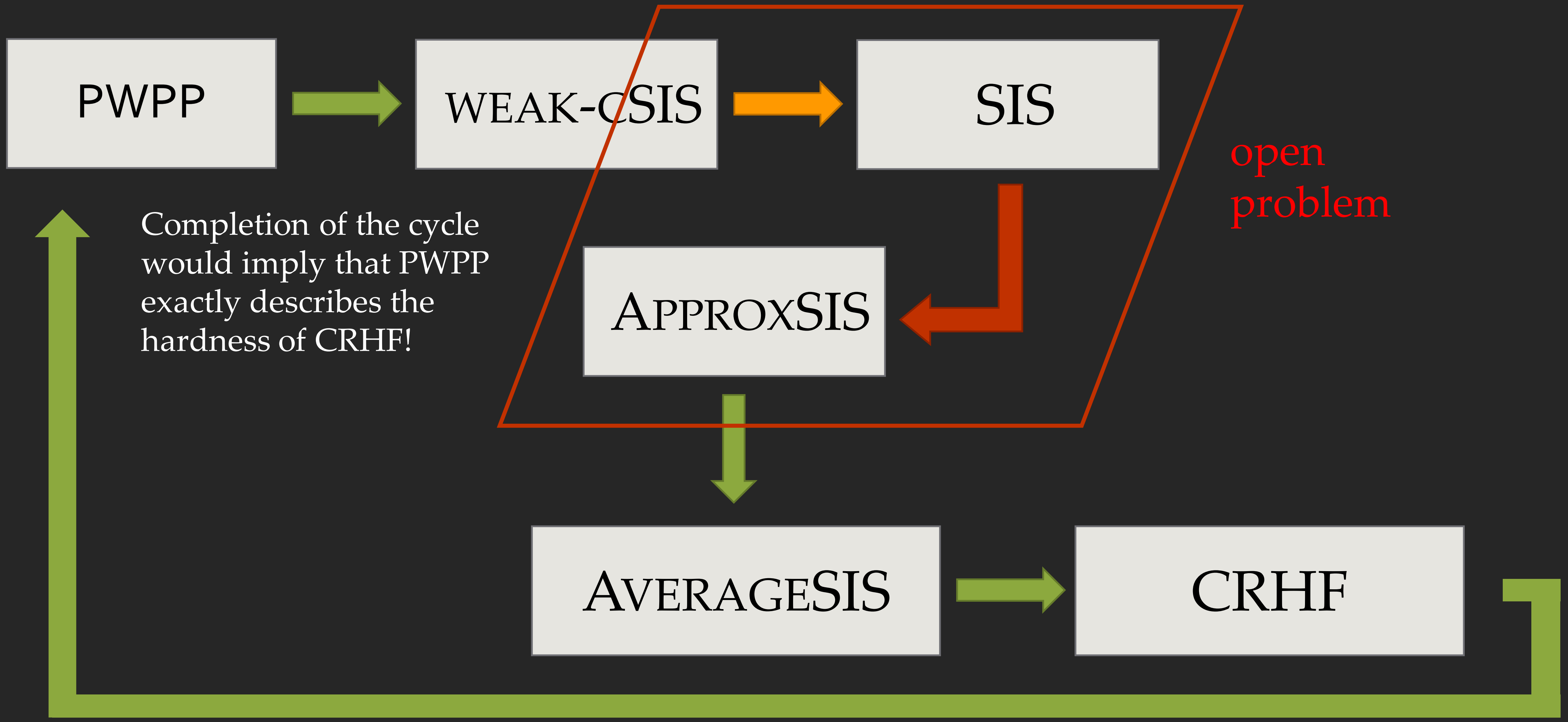
Can we achieve Cryptographic Utopia?



Can we achieve Cryptographic Utopia?



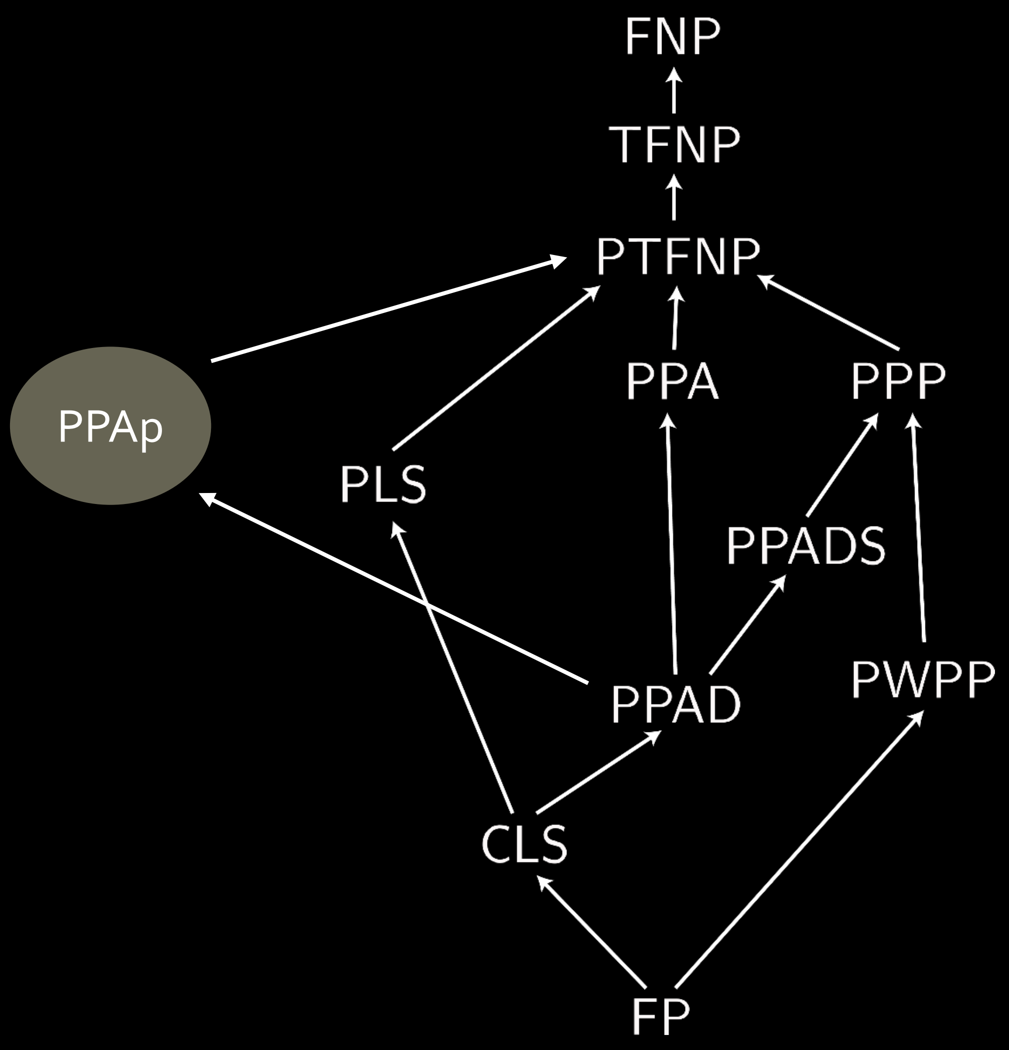
Can we achieve Cryptographic Utopia?



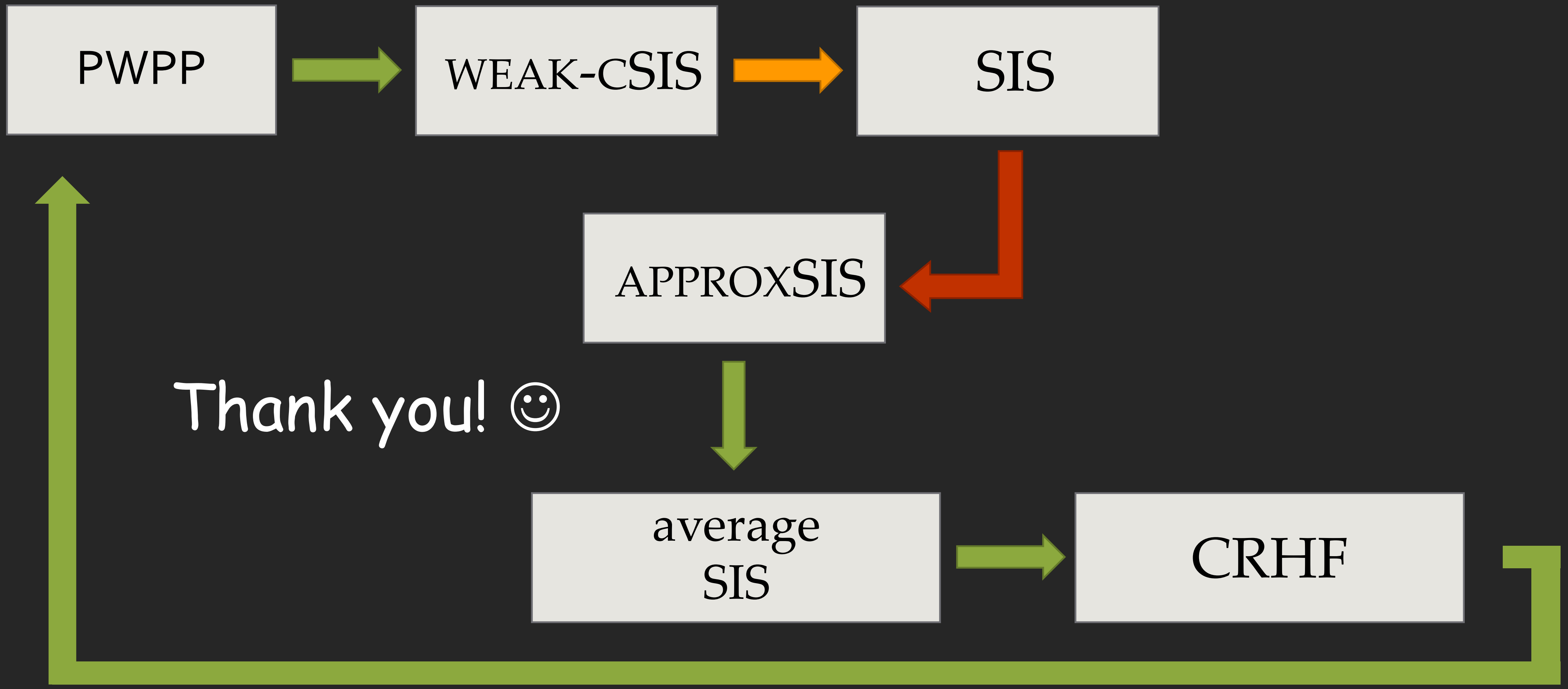
Complexity of Total Search Problems

[Goos Kamath Sotiraki Z.'20]

[Filos-Ratsikas Hollender Sotiraki Z.'20]

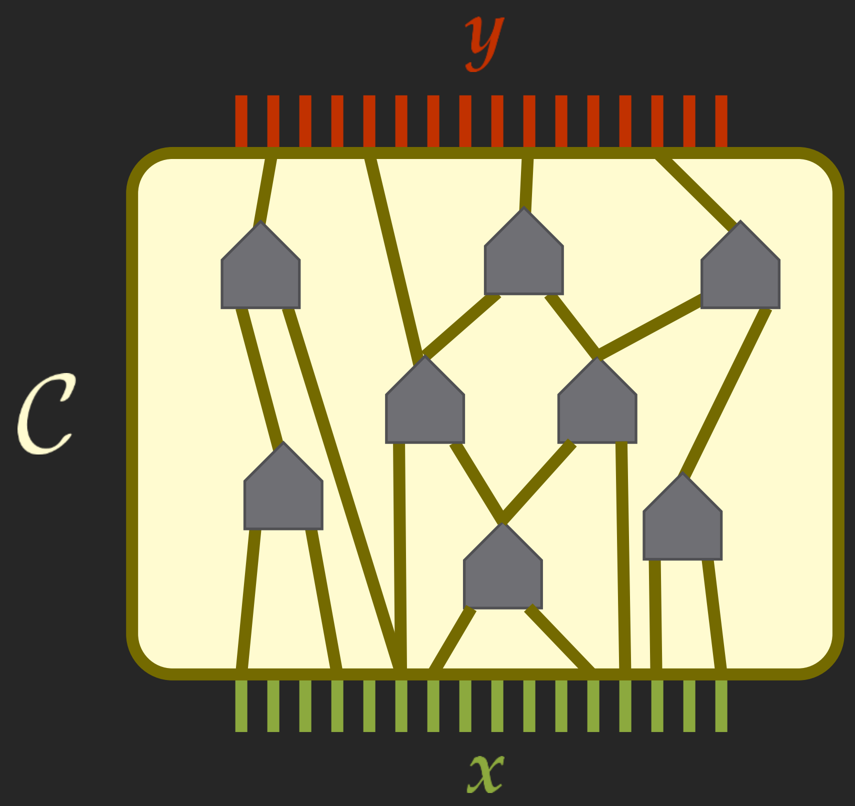


Can we achieve Cryptographic Utopia?



Thank you! ☺

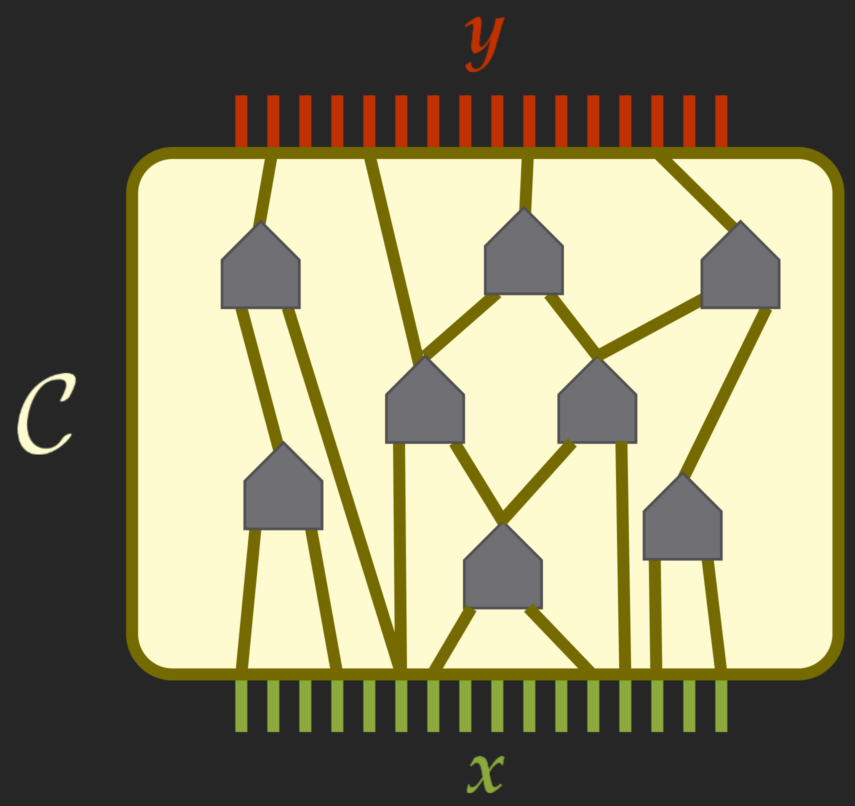
WEAK-cSIS is PWPP-hard

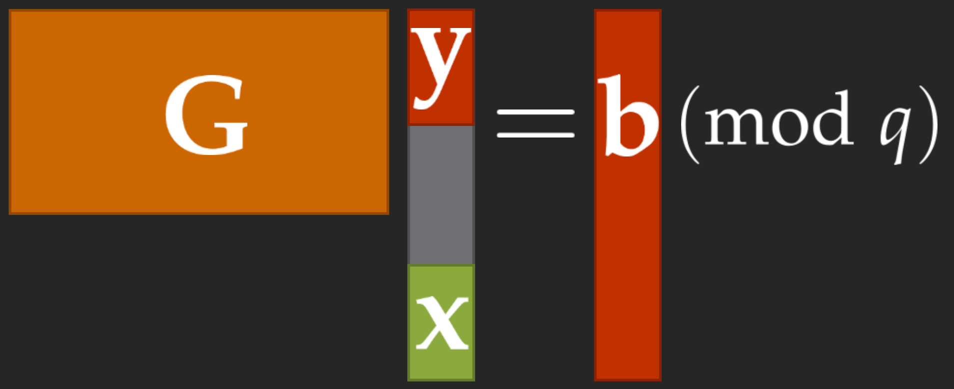


$$\mathbf{G}\begin{bmatrix}\mathbf{y}\\ \\ \mathbf{x}\end{bmatrix} = \mathbf{0} \pmod{q}$$

WEAK-CSIS is PWPP-hard

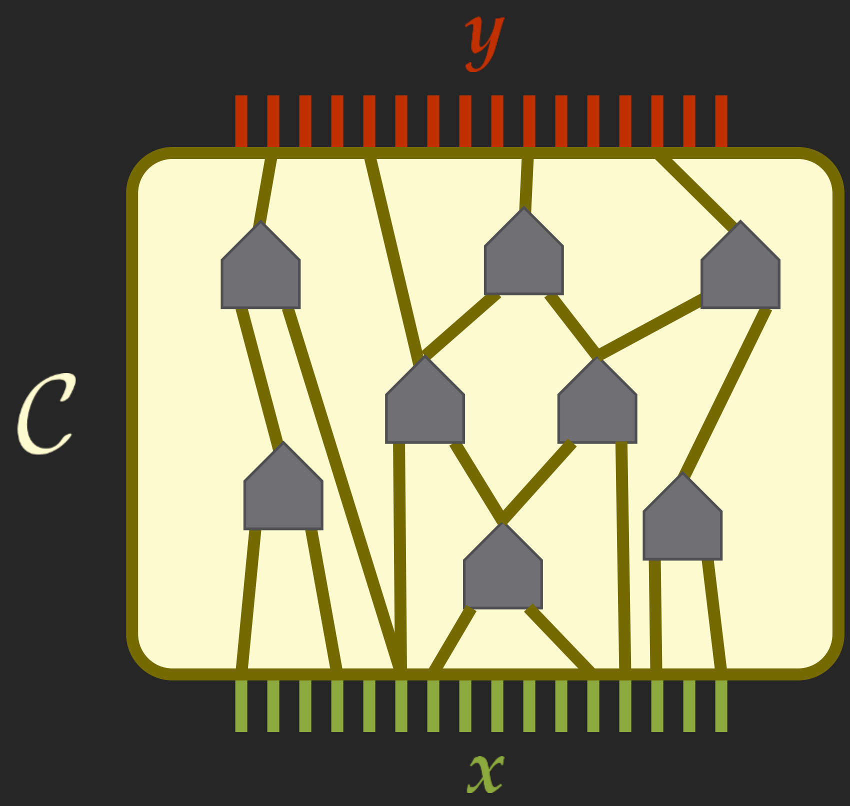
Attention!
During the reduction we have to preserve **totality**!





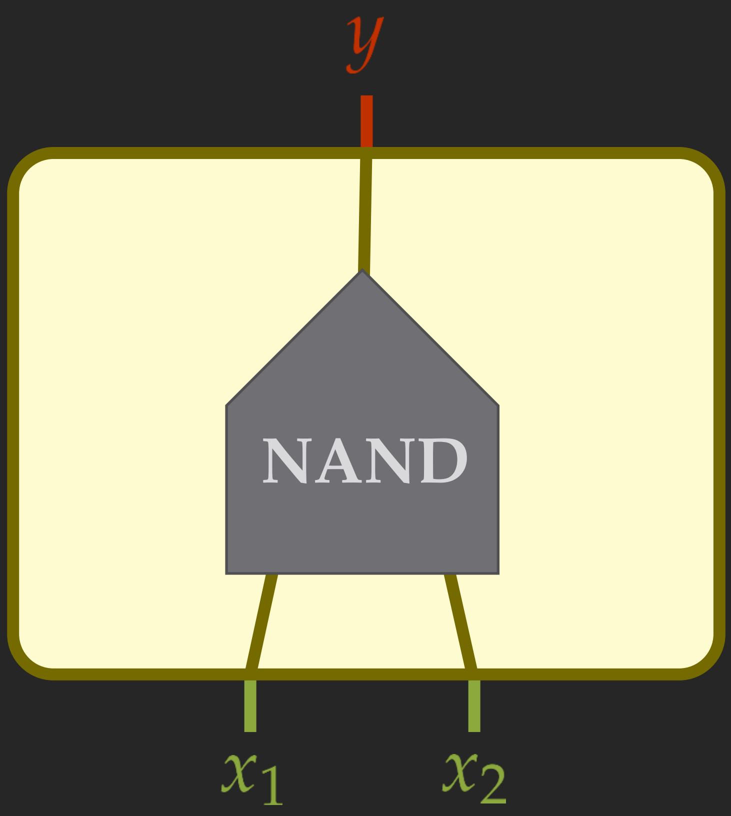
WEAK-cSIS is PWPP-hard

Attention!
During the reduction we have to preserve **totality**!

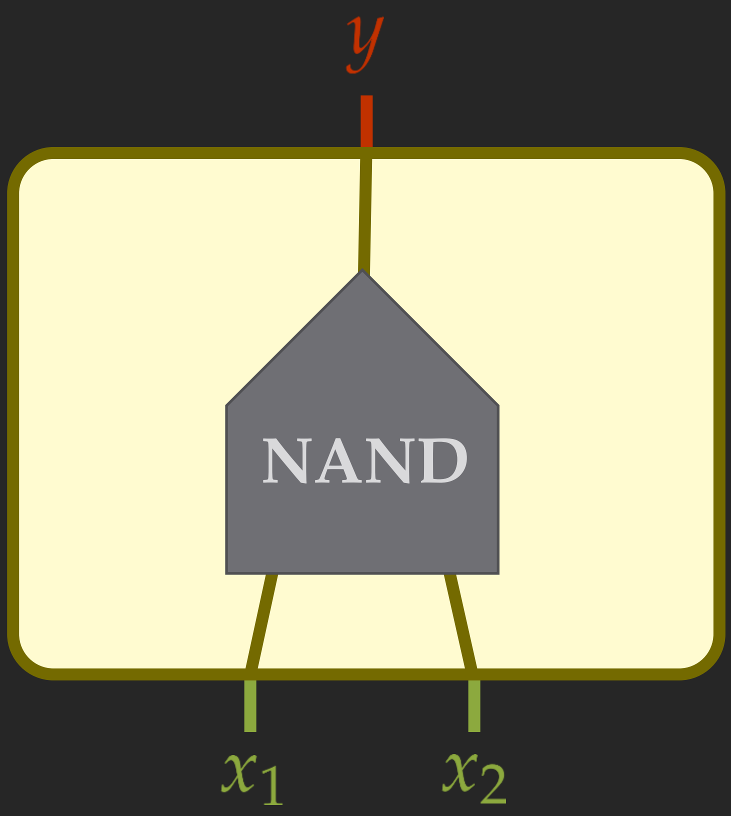


Lemma
For any $\mathbf{z} \in \{0,1\}^{m-\log(q)d}$ and any $\mathbf{b} \in \mathbb{Z}_1^d$, we can efficiently compute a binary solution of the form $\mathbf{x} = [\star \ \ \star \cdots \star \ \ \mathbf{z}]$ for the equation $\mathbf{Gx} = \mathbf{b} \pmod q$.

WEAK-cSIS is PWPP-hard

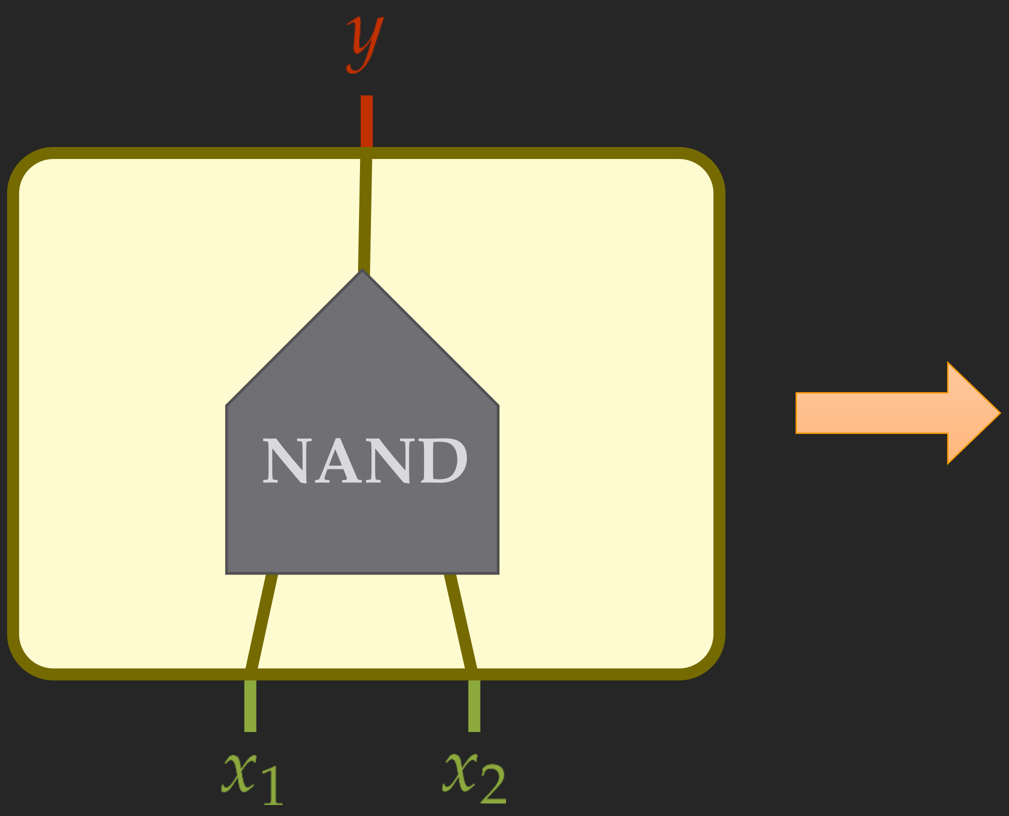


WEAK-cSIS is PWPP-hard



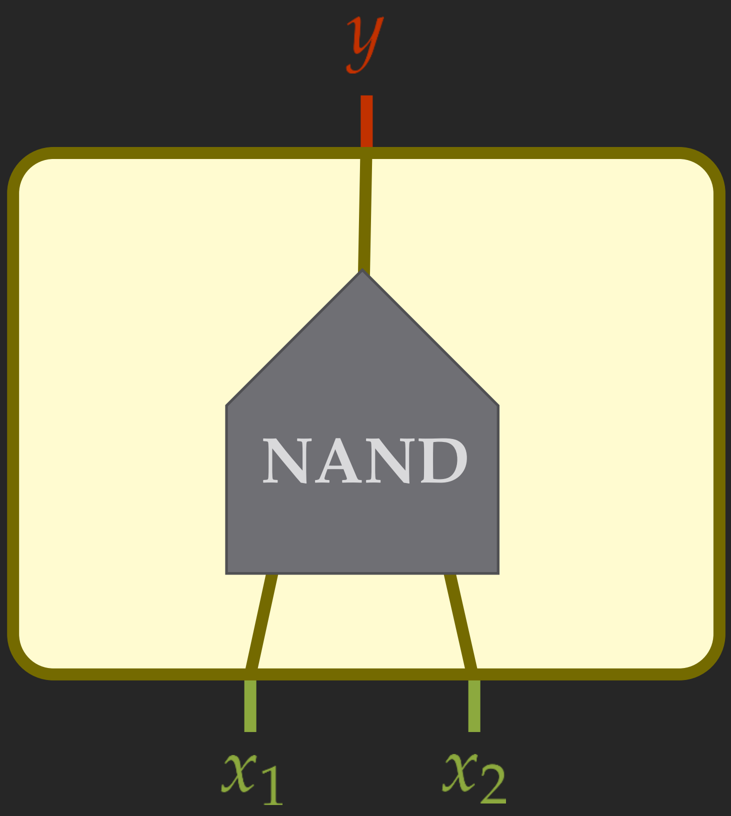
$$1 \cdot v + 2 \cdot y - x_1 - x_2 = 2 \pmod 4$$

WEAK-cSIS is PWPP-hard



$$\underbrace{\boxed{1 \cdot v + 2 \cdot y}}_{\mathbf{g}} - x_1 - x_2 = 2 \pmod 4$$

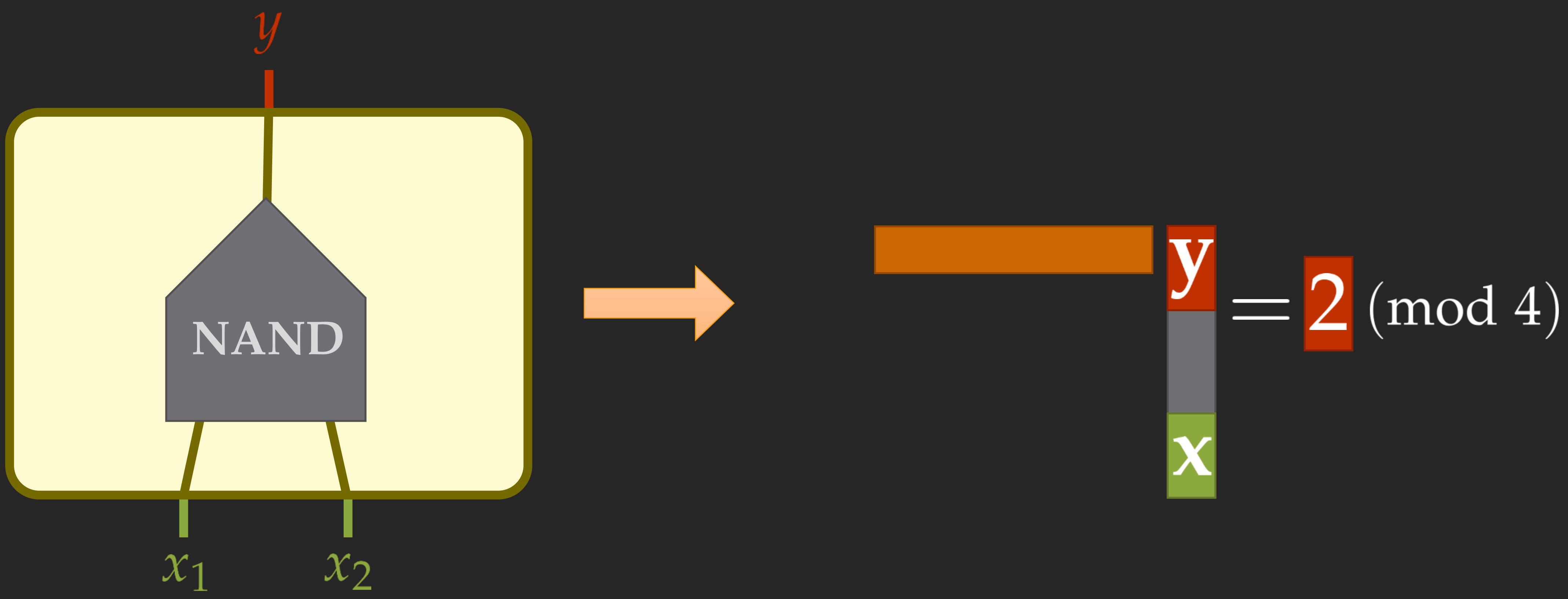
WEAK-cSIS is PWPP-hard



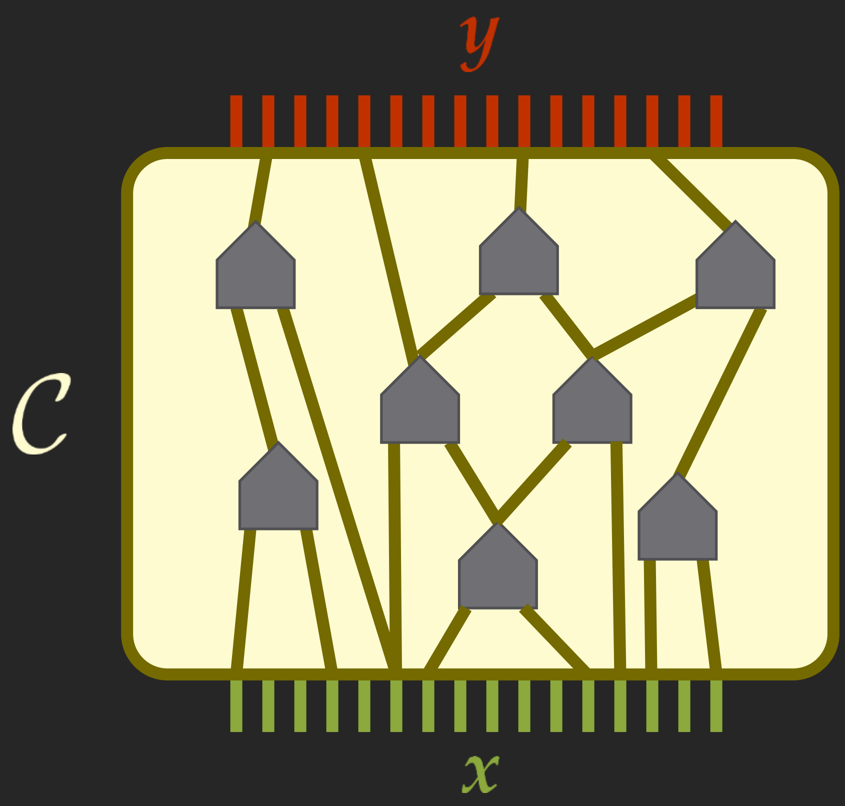
$$1 \cdot v + 2 \cdot y - x_1 - x_2 = 2 \pmod 4$$

0	1	0	0
1	1	0	1
1	1	1	0
0	0	1	1

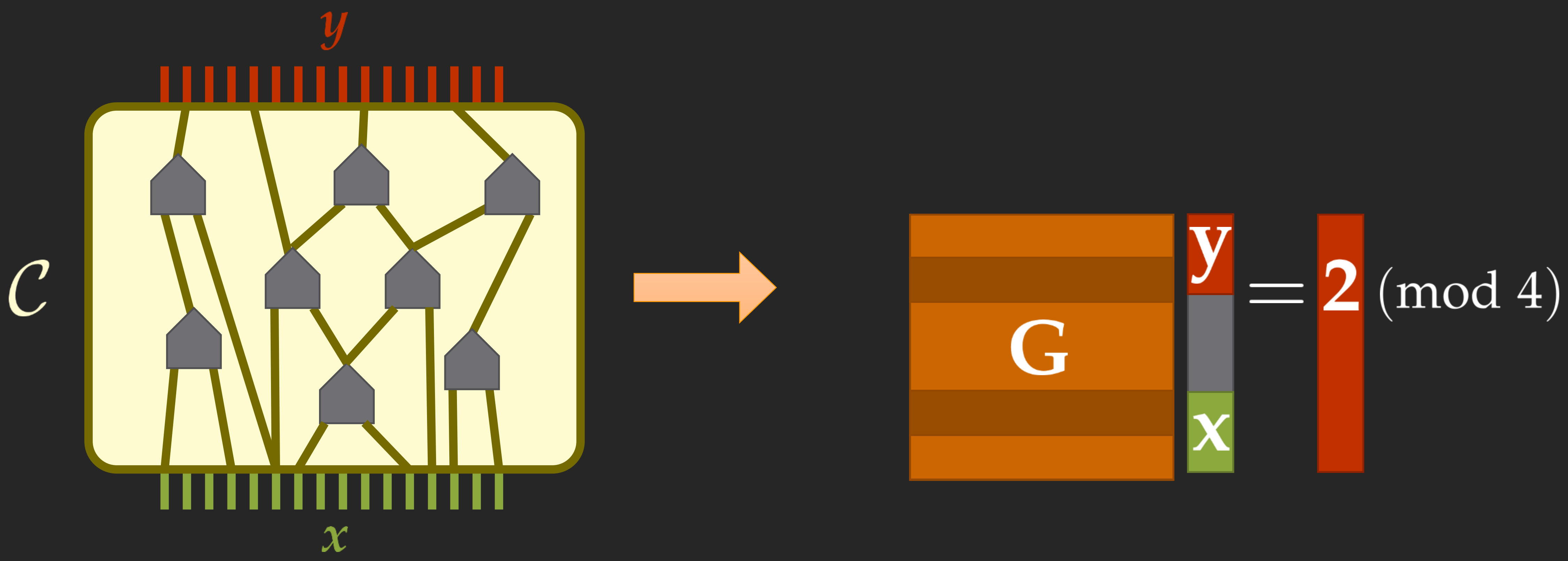
WEAK-cSIS is PWPP-hard



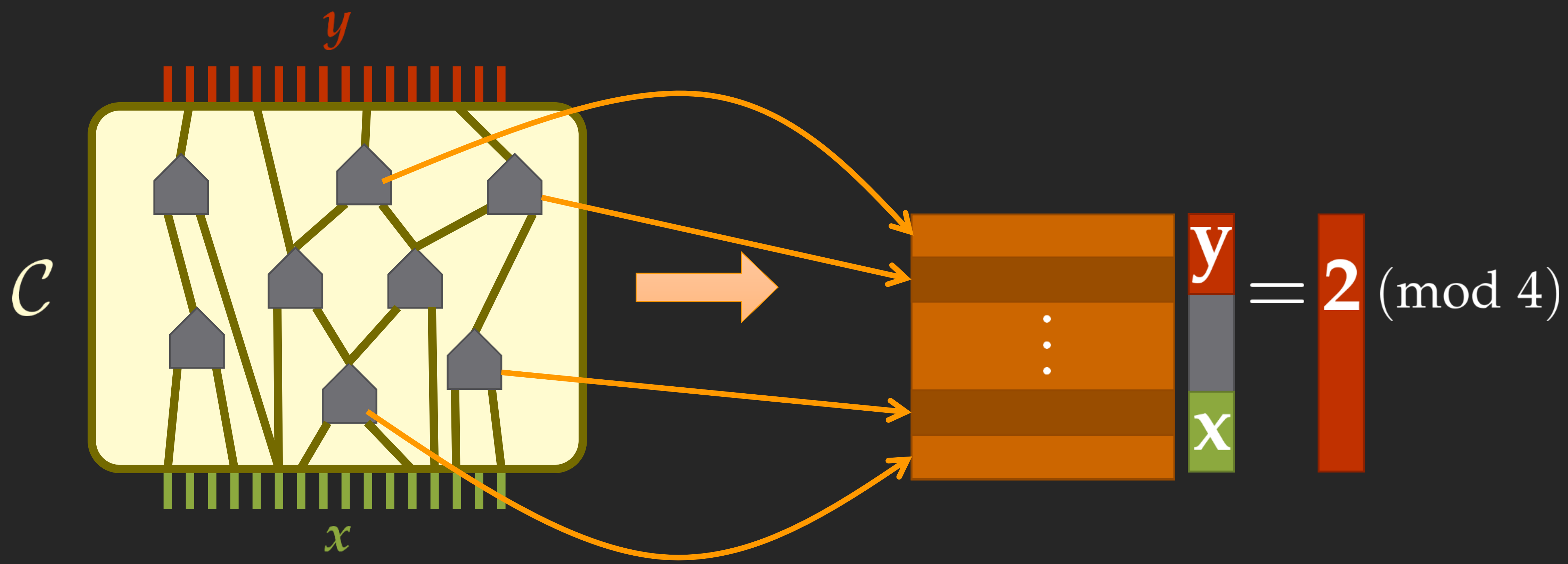
WEAK-cSIS is PWPP-hard



WEAK-cSIS is PWPP-hard

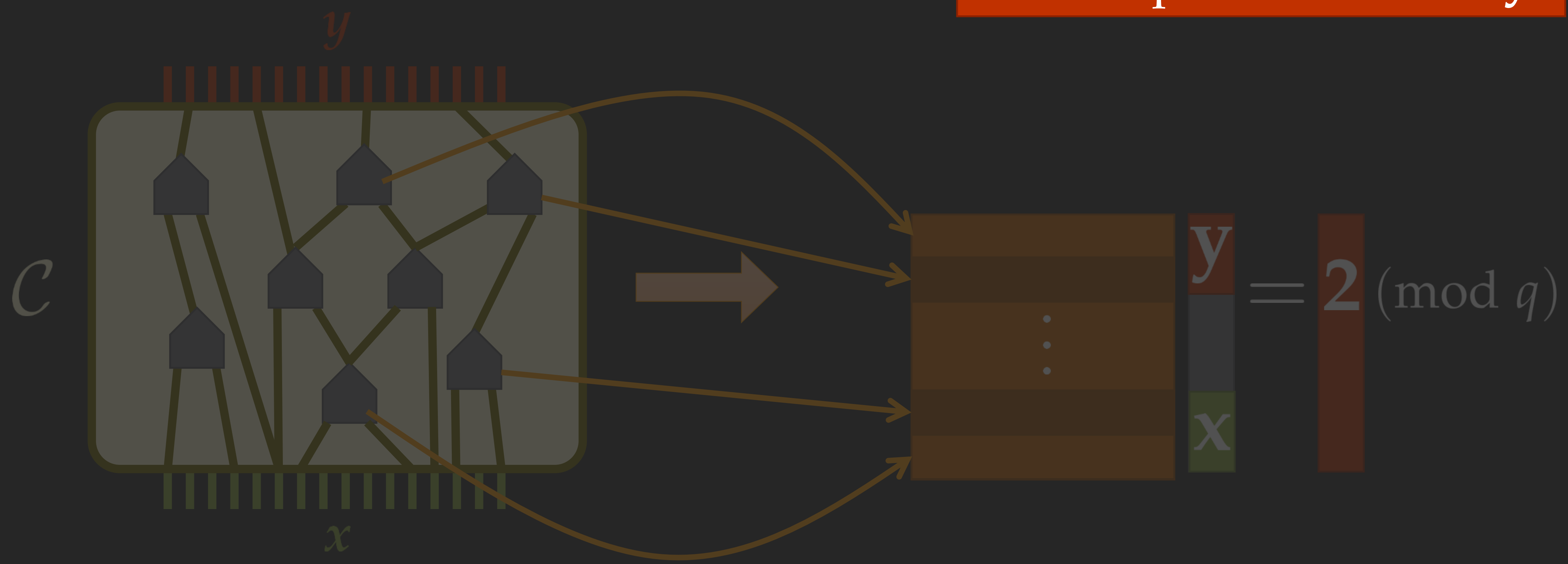


WEAK-cSIS is PWPP-hard



WEAK-cSIS is PWPP-hard

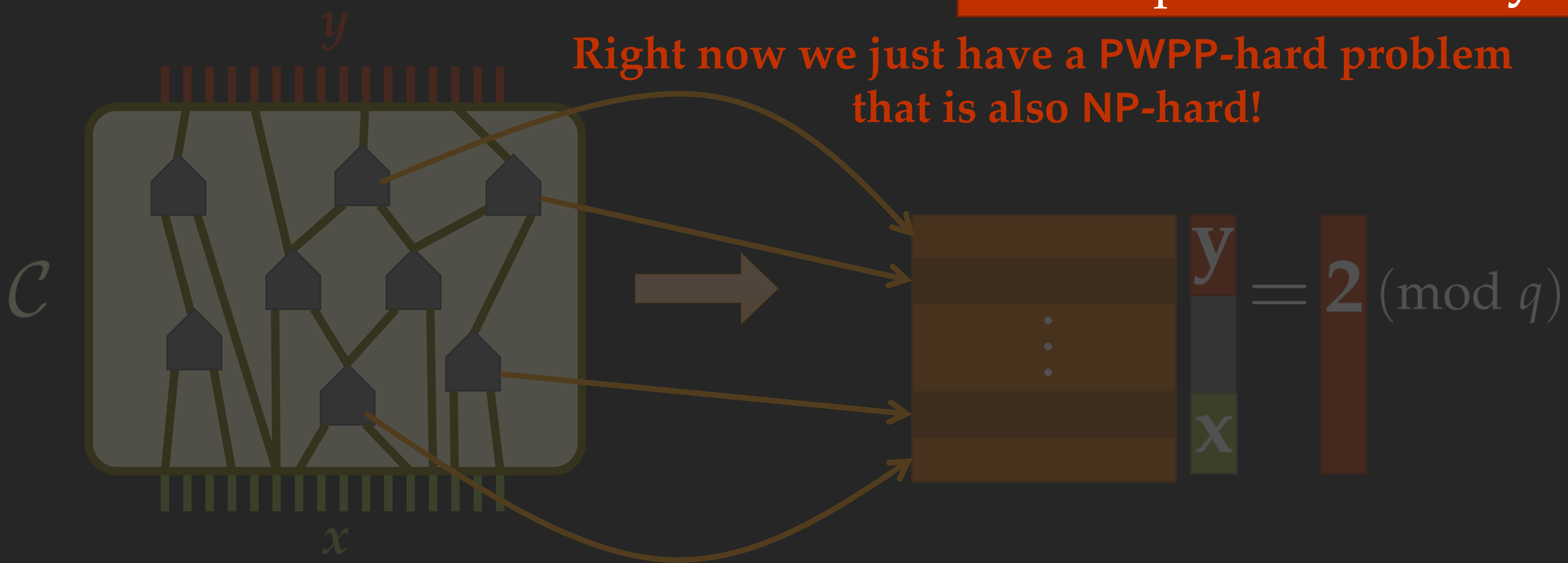
Attention!
During the reduction we have to preserve **totality**!



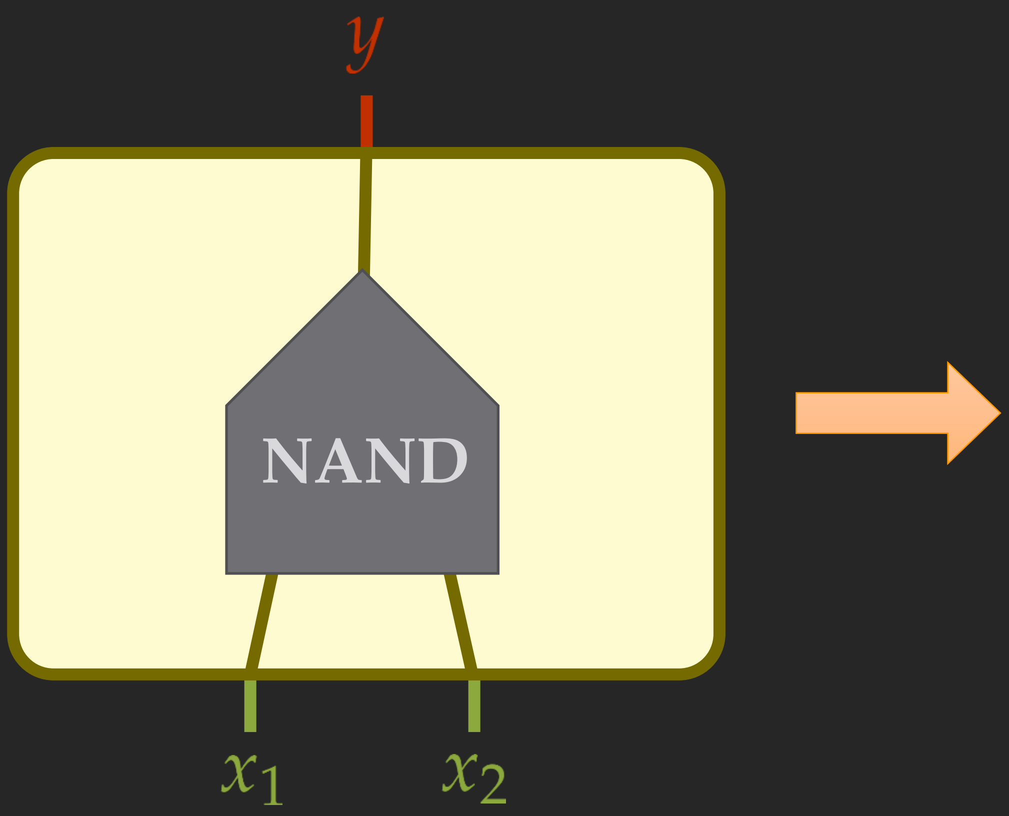
WEAK-CSIS is PWPP-hard

Attention!
During the reduction we have to preserve **totality**!

Right now we just have a PWPP-hard problem that is also NP-hard!

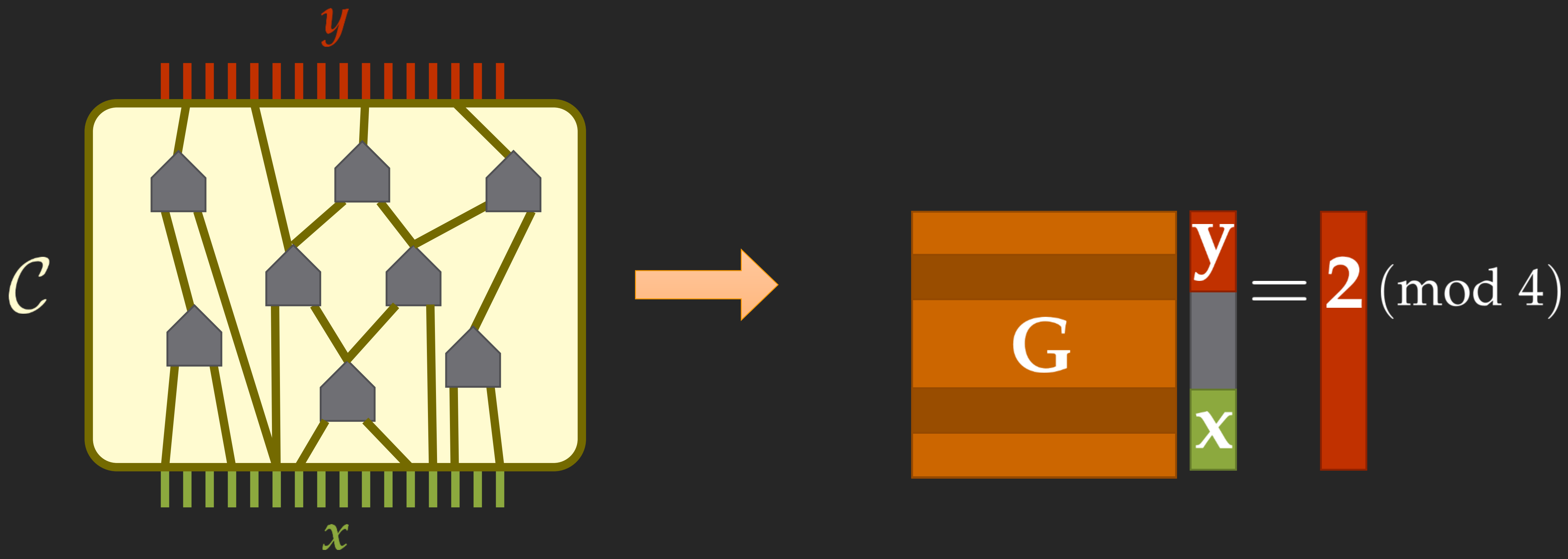


WEAK-cSIS is PWPP-hard

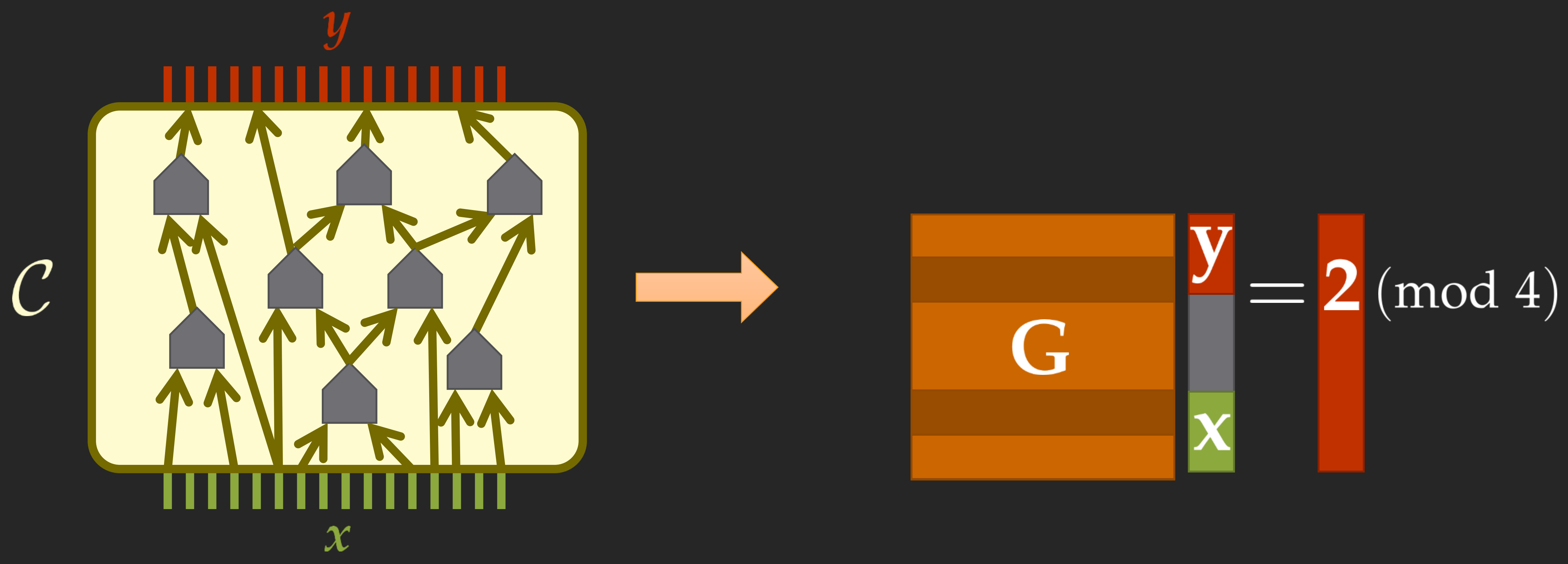


$$\underbrace{\boxed{1 \cdot v + 2 \cdot y}}_{\mathbf{g}} - x_1 - x_2 = 2 \pmod 4$$

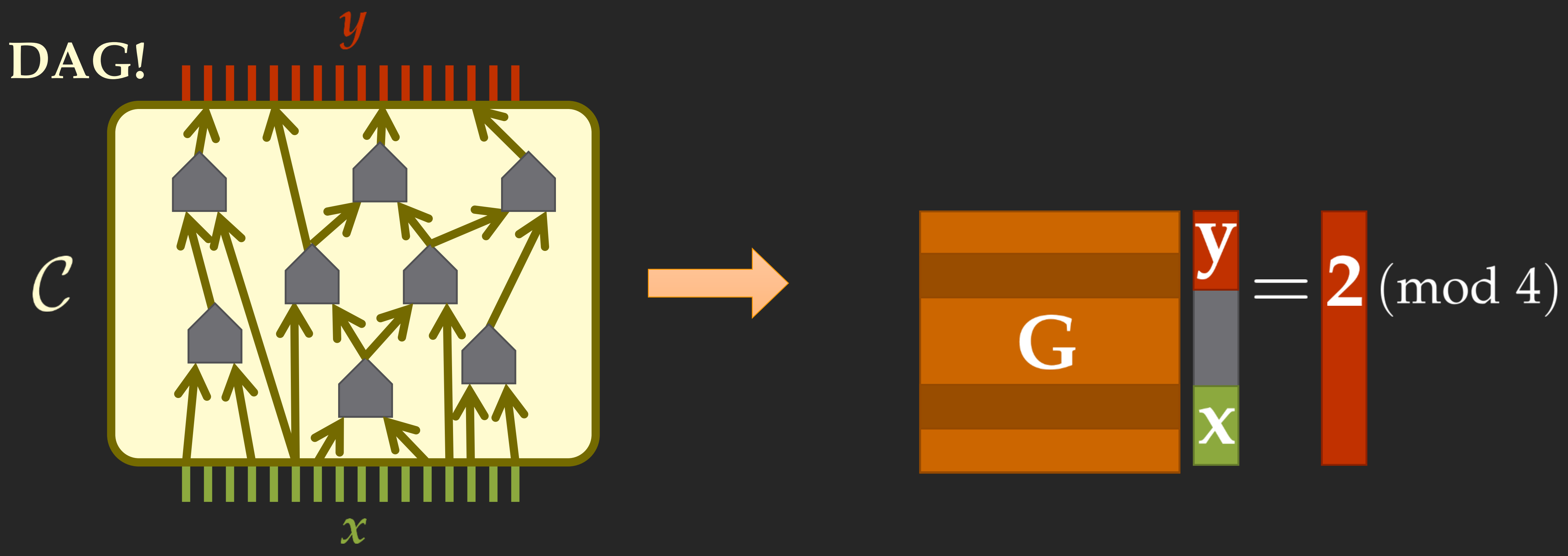
WEAK-cSIS is PWPP-hard



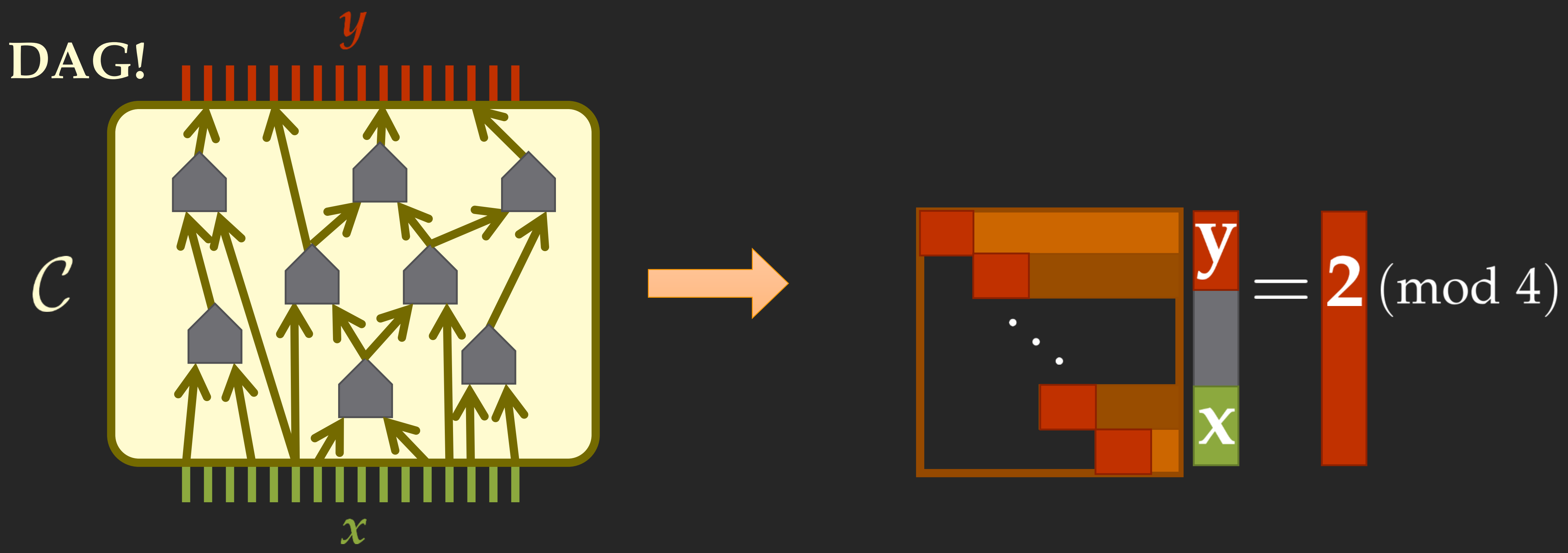
WEAK-cSIS is PWPP-hard



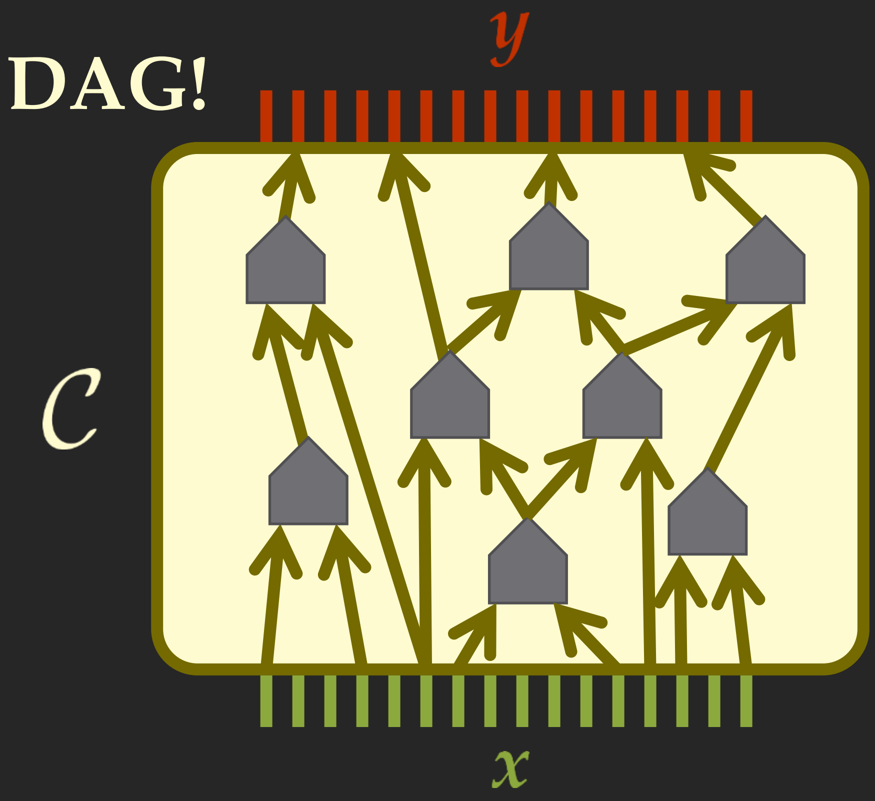
WEAK-CSIS is PWPP-hard

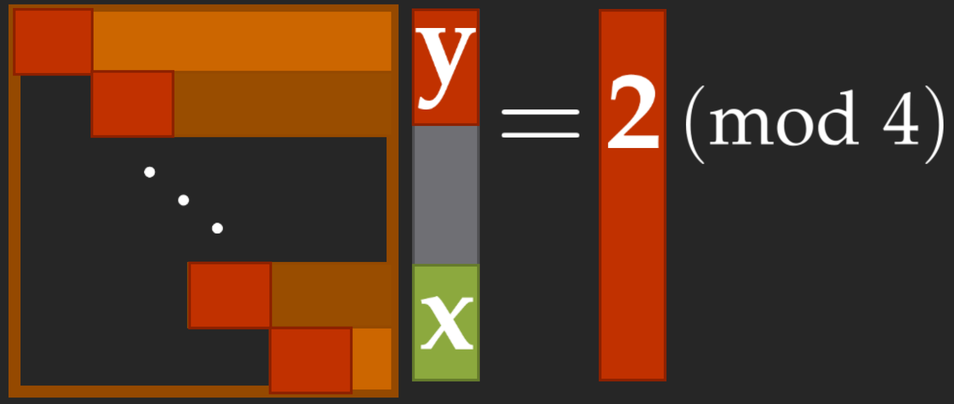


WEAK-cSIS is PWPP-hard



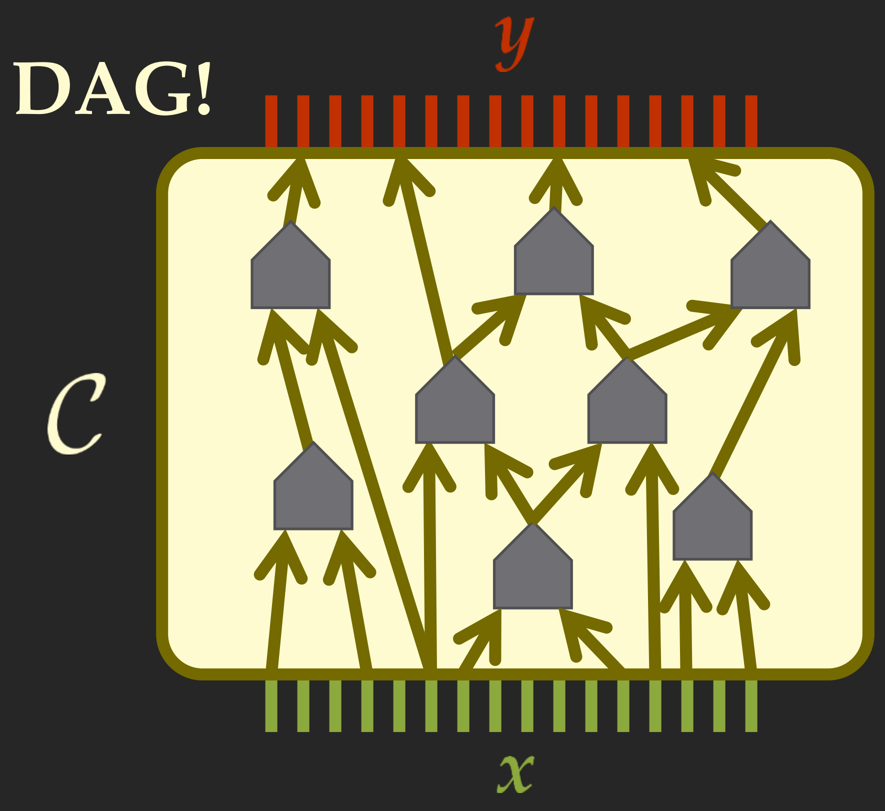
WEAK-cSIS is PWPP-hard

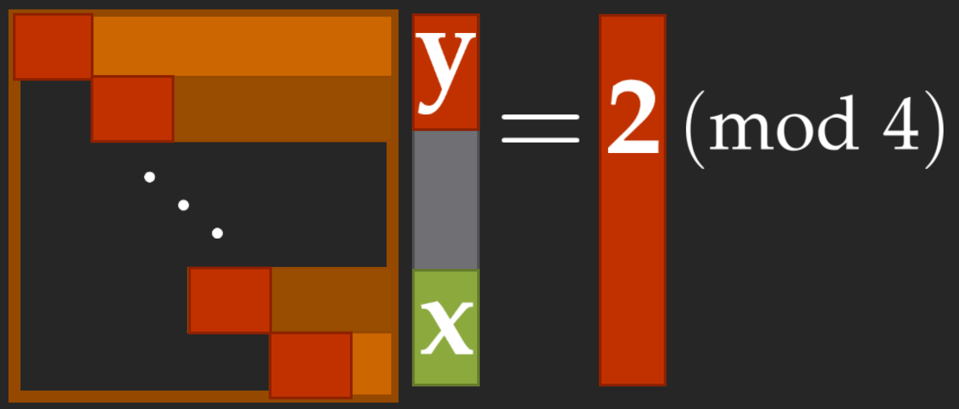




Binary invertible!

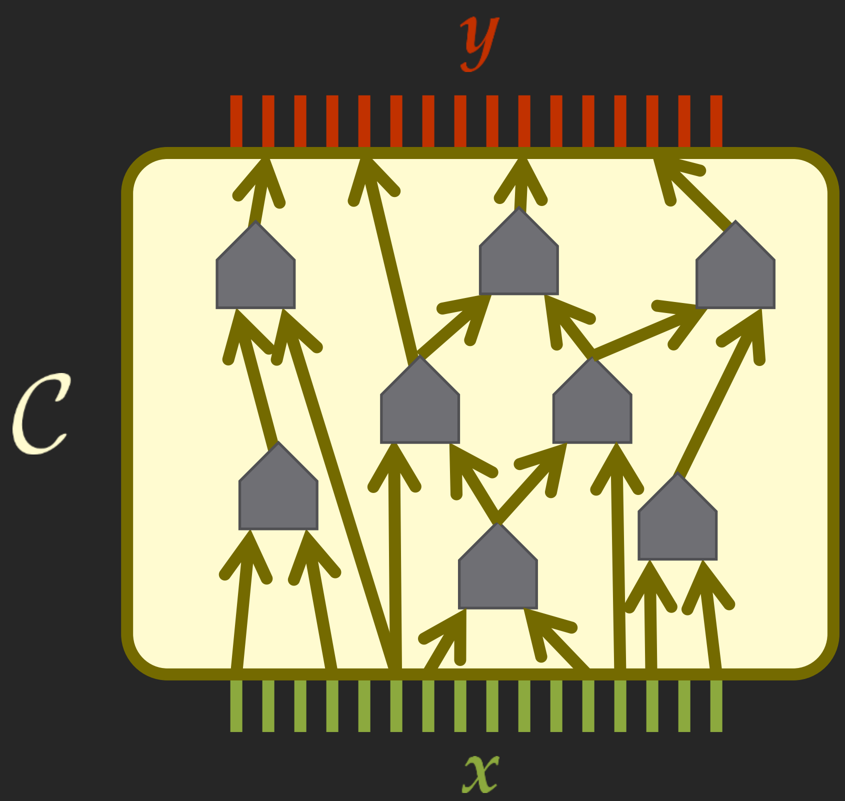
WEAK-cSIS is PWPP-hard

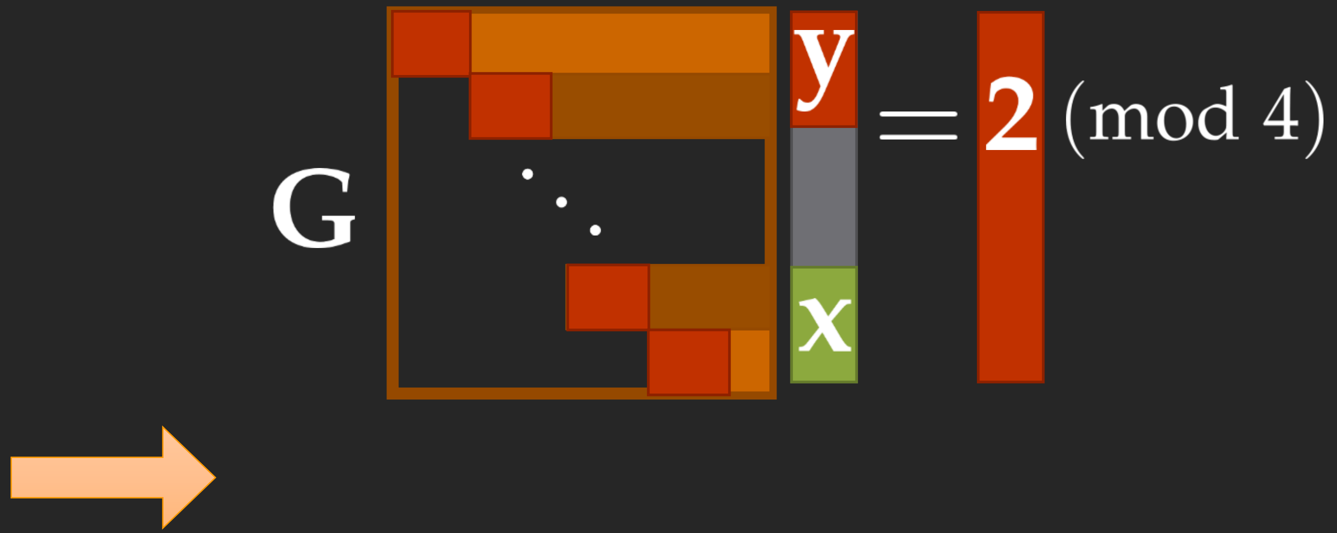




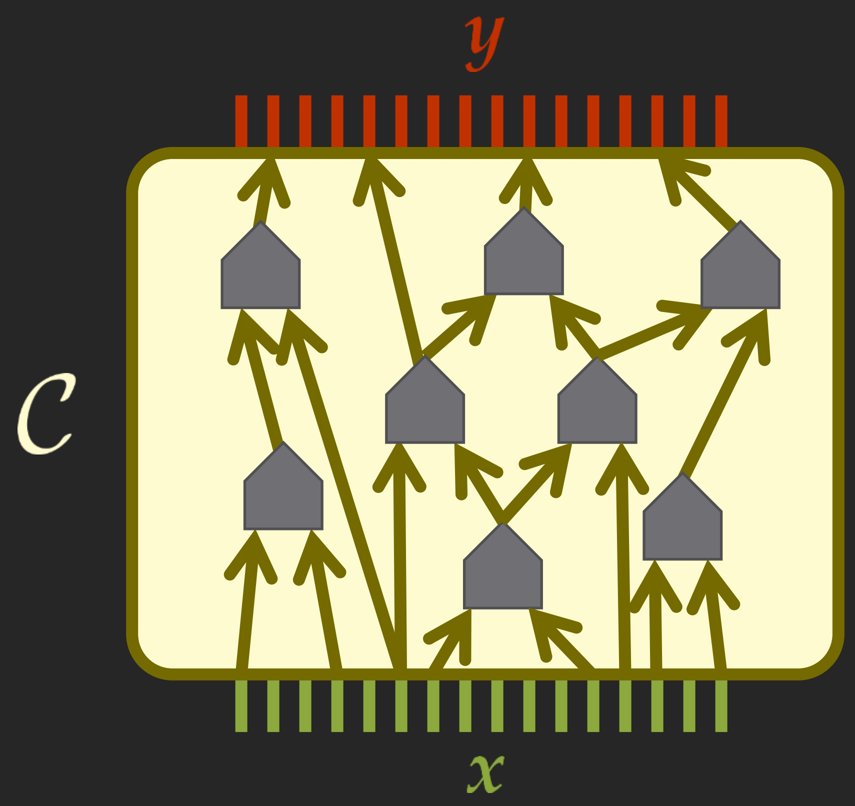
Binary invertible!

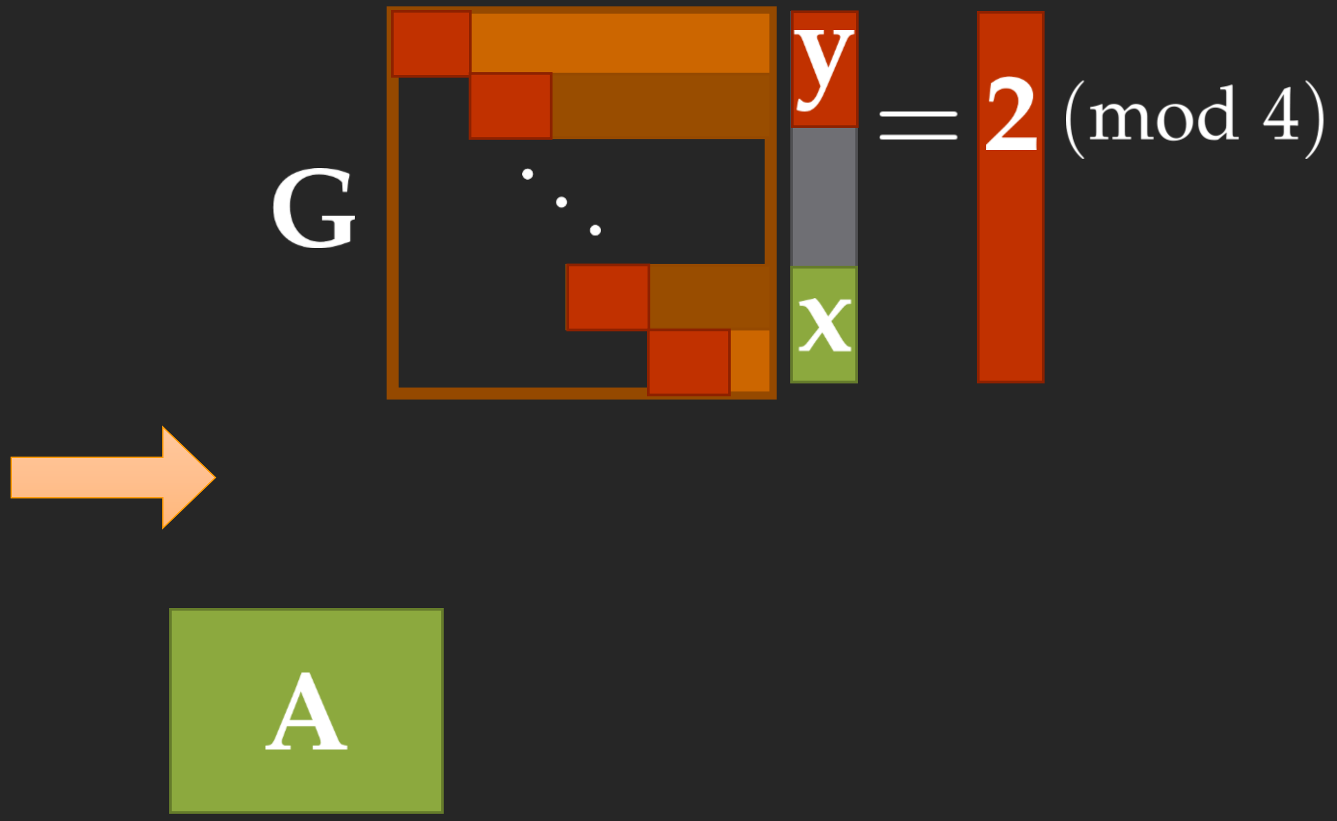
WEAK-cSIS is PWPP-hard



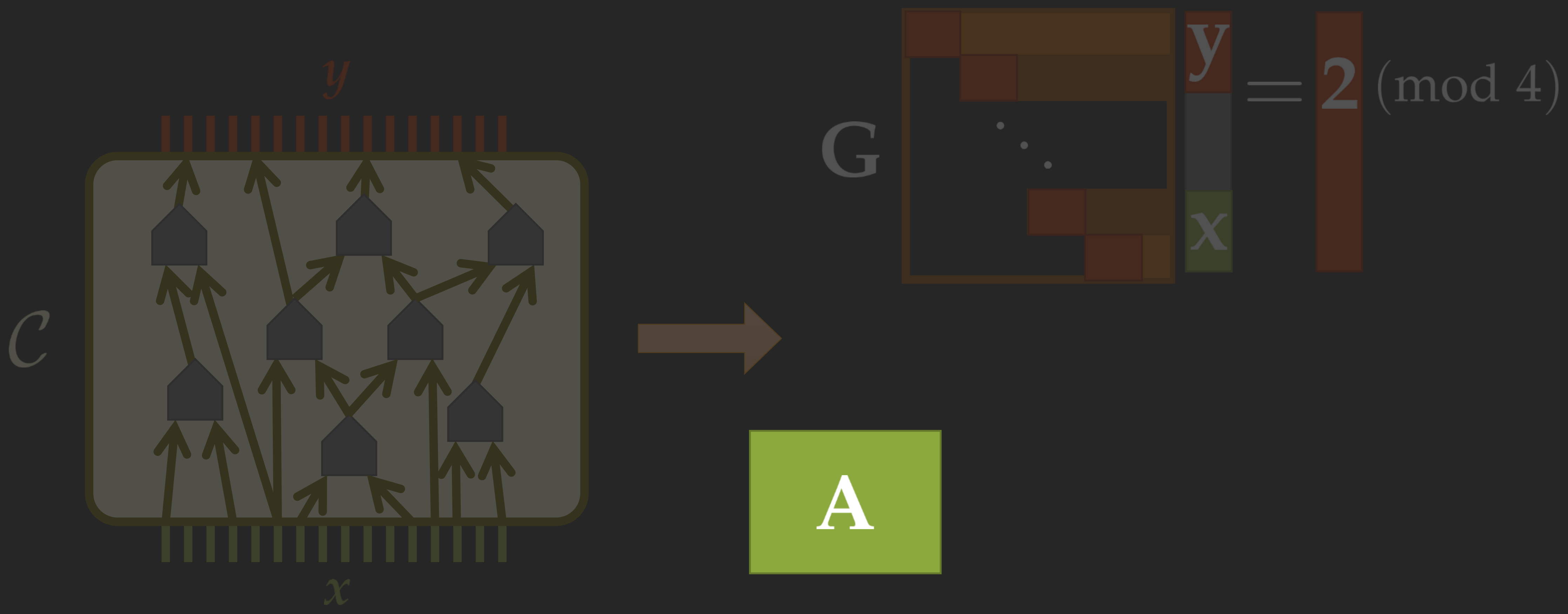


WEAK-cSIS is PWPP-hard

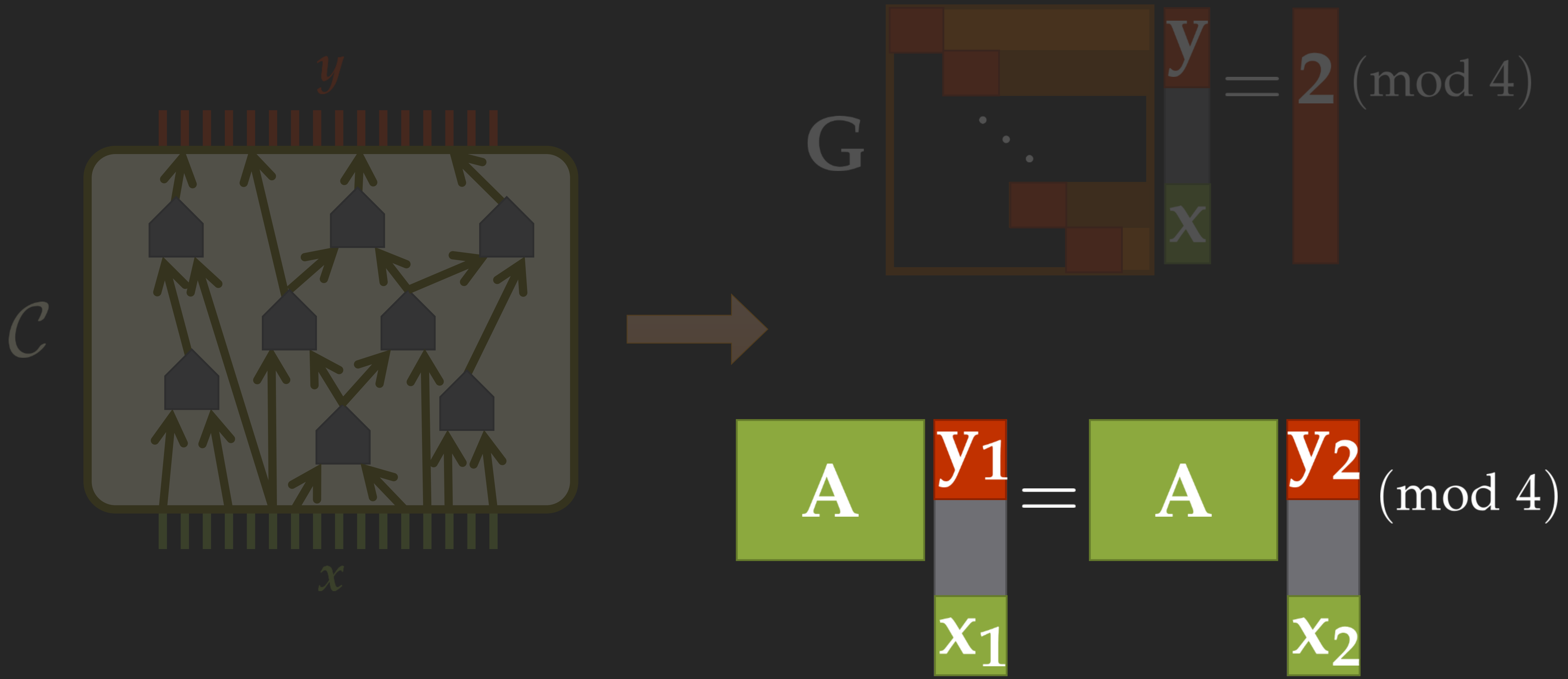




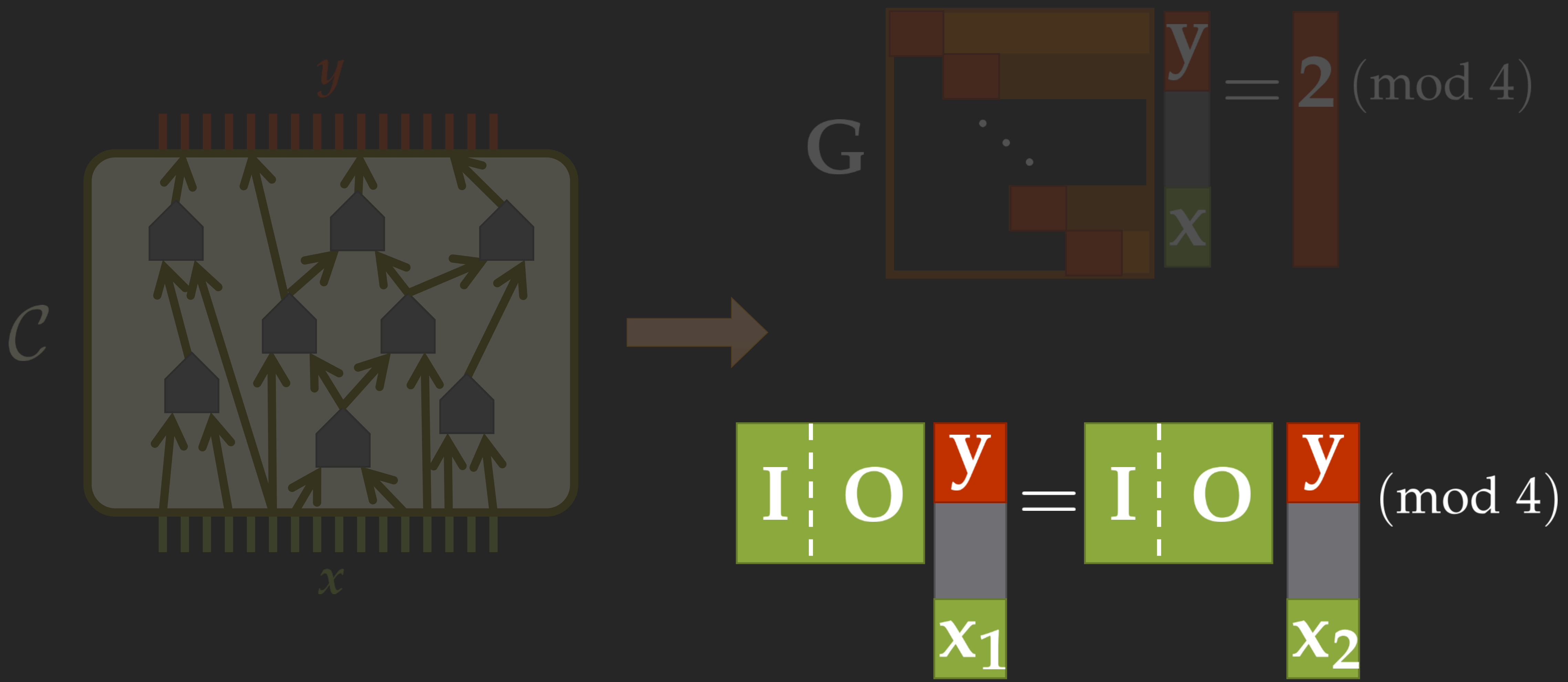
WEAK-cSIS is PWPP-hard



WEAK-cSIS is PWPP-hard

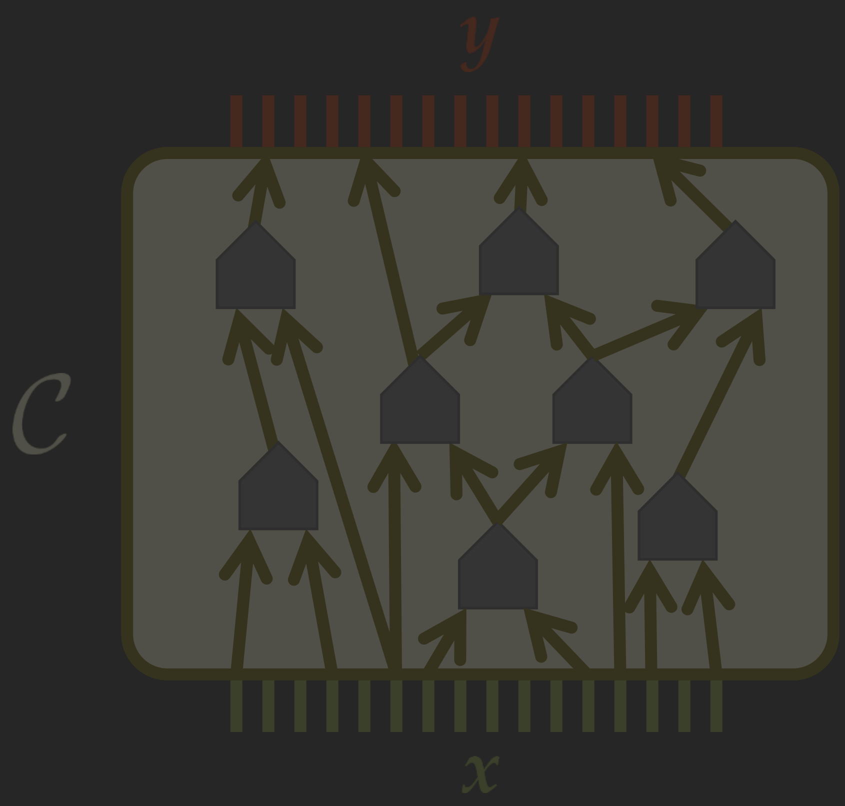


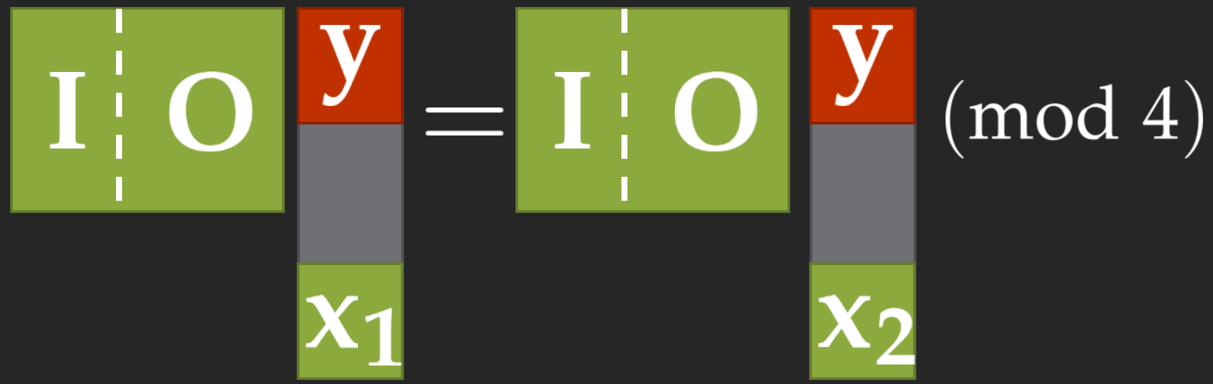
WEAK-cSIS is PWPP-hard



WEAK-cSIS is PWPP-hard

Attention!
During the reduction we have to preserve **totality**!





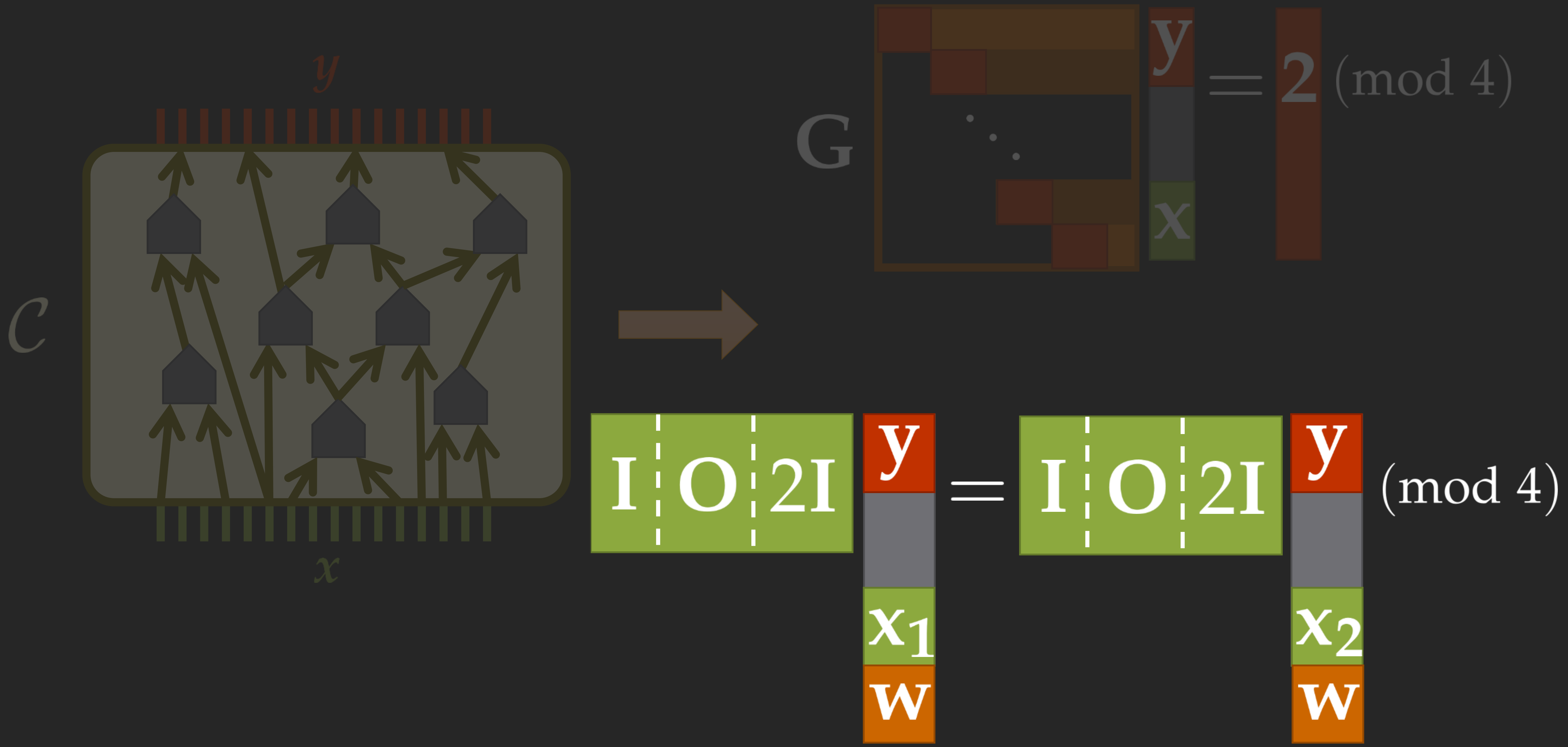
WEAK-CSIS is PWPP-hard

INPUT: $\mathbf{A} \in \mathbb{Z}_q^{r\times m}$, with $m > \log(q)(r+d)$ $\mathbf{G} \in \mathbb{Z}_q^{d\times m}$, and *binary invertible*

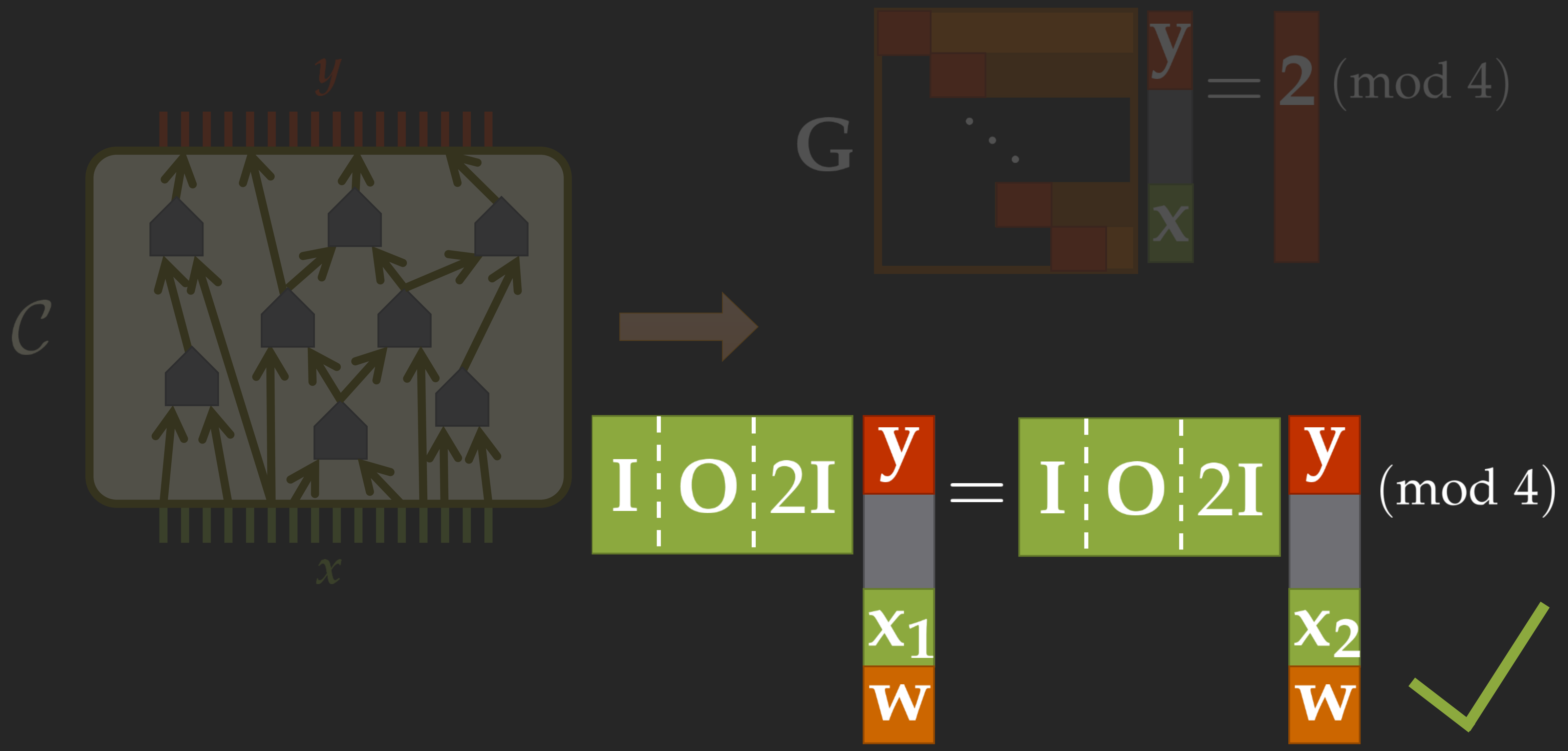
OUTPUT: $\mathbf{x}\ \mathbf{y} \in \{0,1\}^m$ s.t. $\mathbf{A}\mathbf{x} = \mathbf{A}\mathbf{y} \pmod q$

$\mathbf{G}\mathbf{x} = \mathbf{G}\mathbf{y} = \mathbf{0} \pmod q$

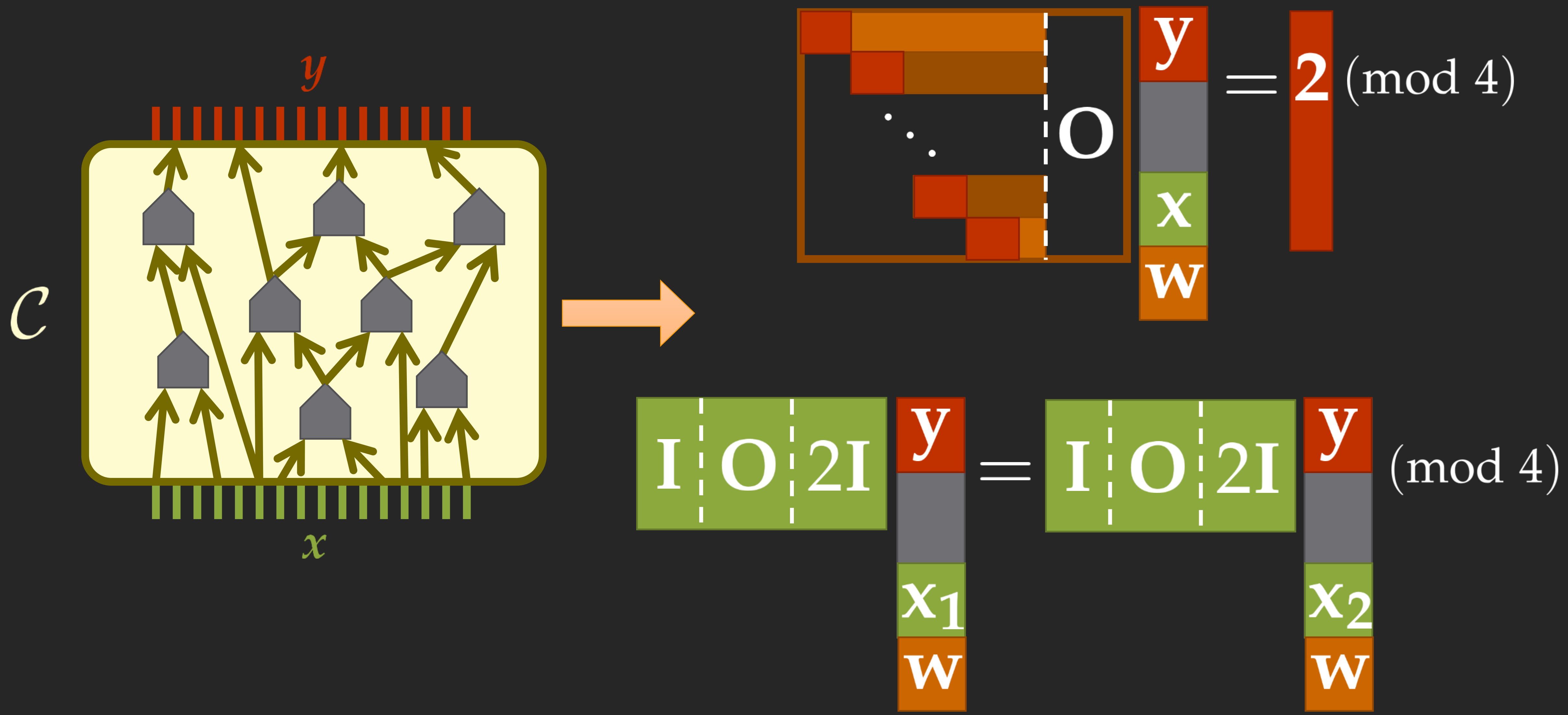
WEAK-CSIS is PWPP-hard



WEAK-cSIS is PWPP-hard



WEAK-CSIS is PWPP-hard



WEAK-cSIS is PWPP-hard