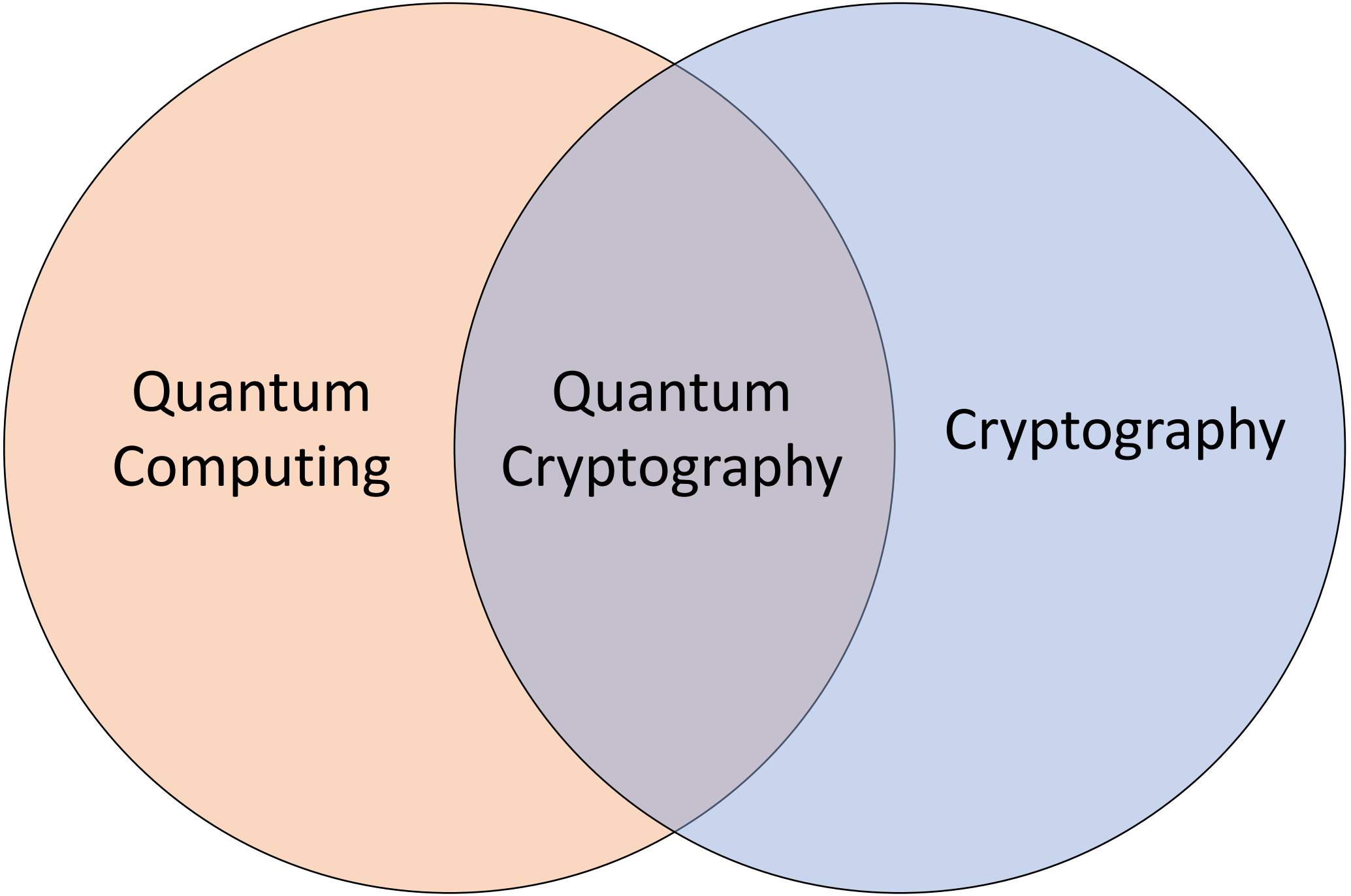
The Surprising Power of Quantum Cryptography

Mark Zhandry
NTT Research


Quantum
Computing
Quantum
Cryptography
Cryptography


Cryptography is everywhere

Modern cryptography is built on ***provable*** security

Provably secure applications

Significant community cryptanalysis effort

Factoring

Discrete logarithms

SHA2

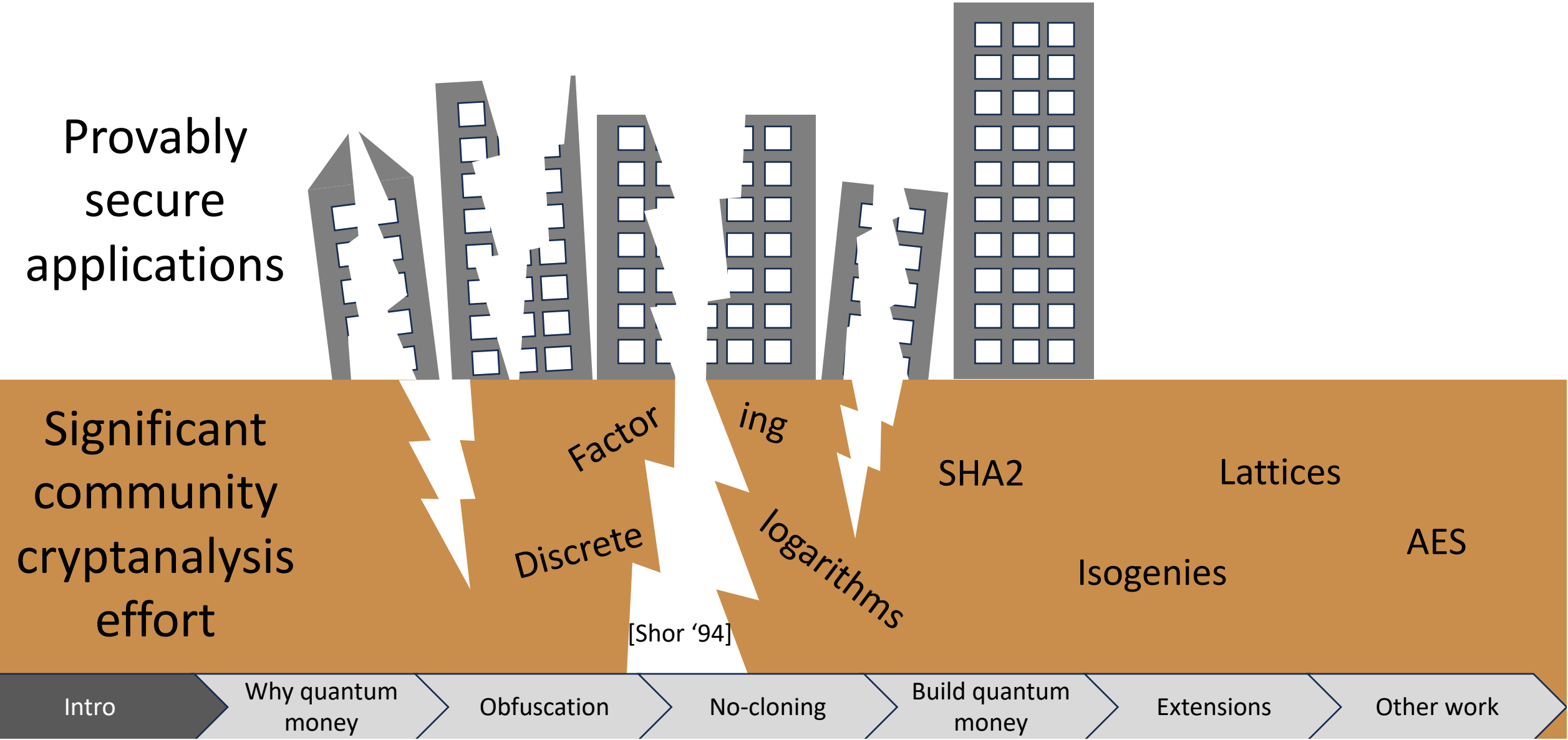
Isogenies

Lattices

AES

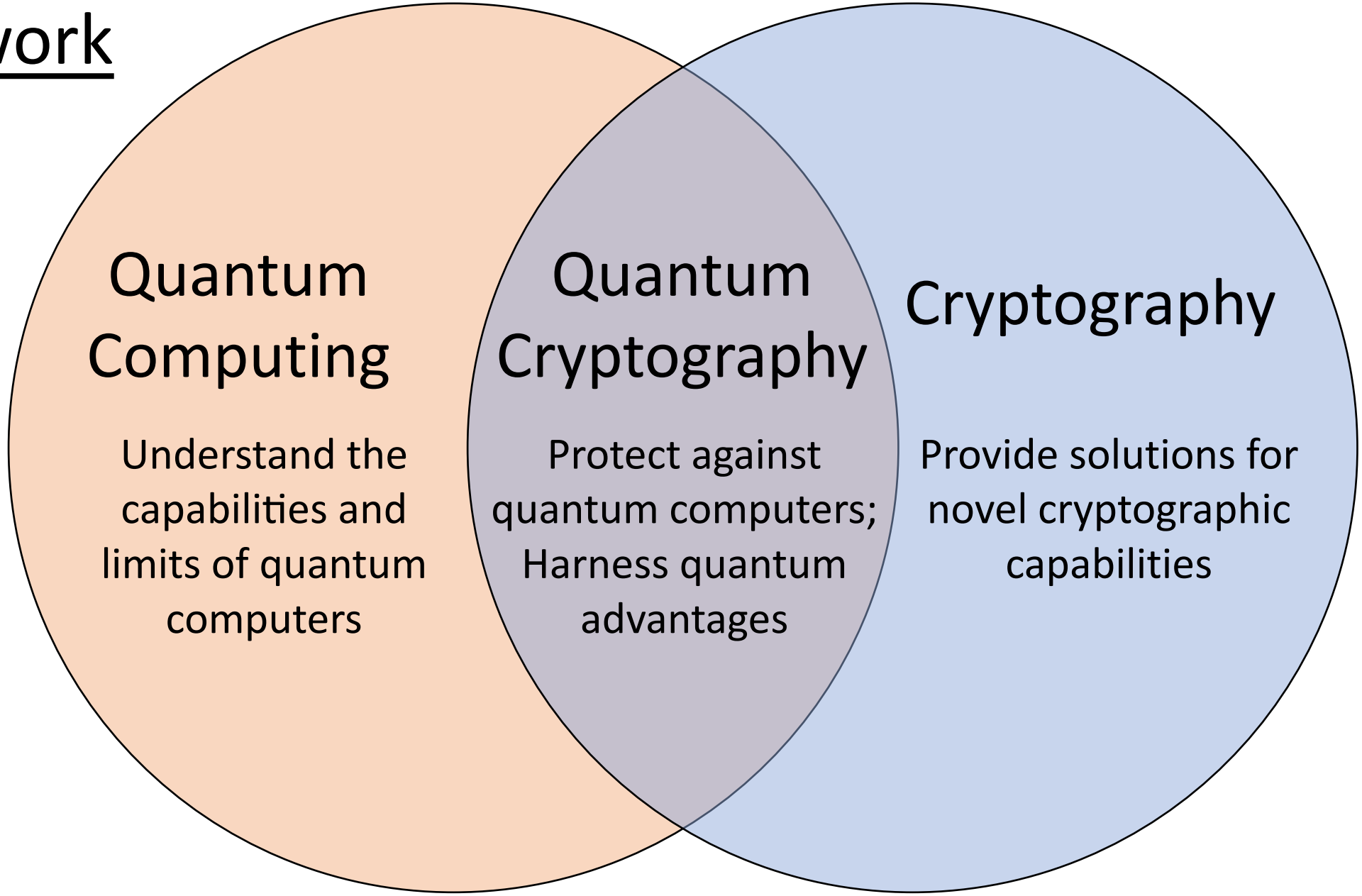
Quantum computers will change everything

Quantum computers pose threats and new possibilities

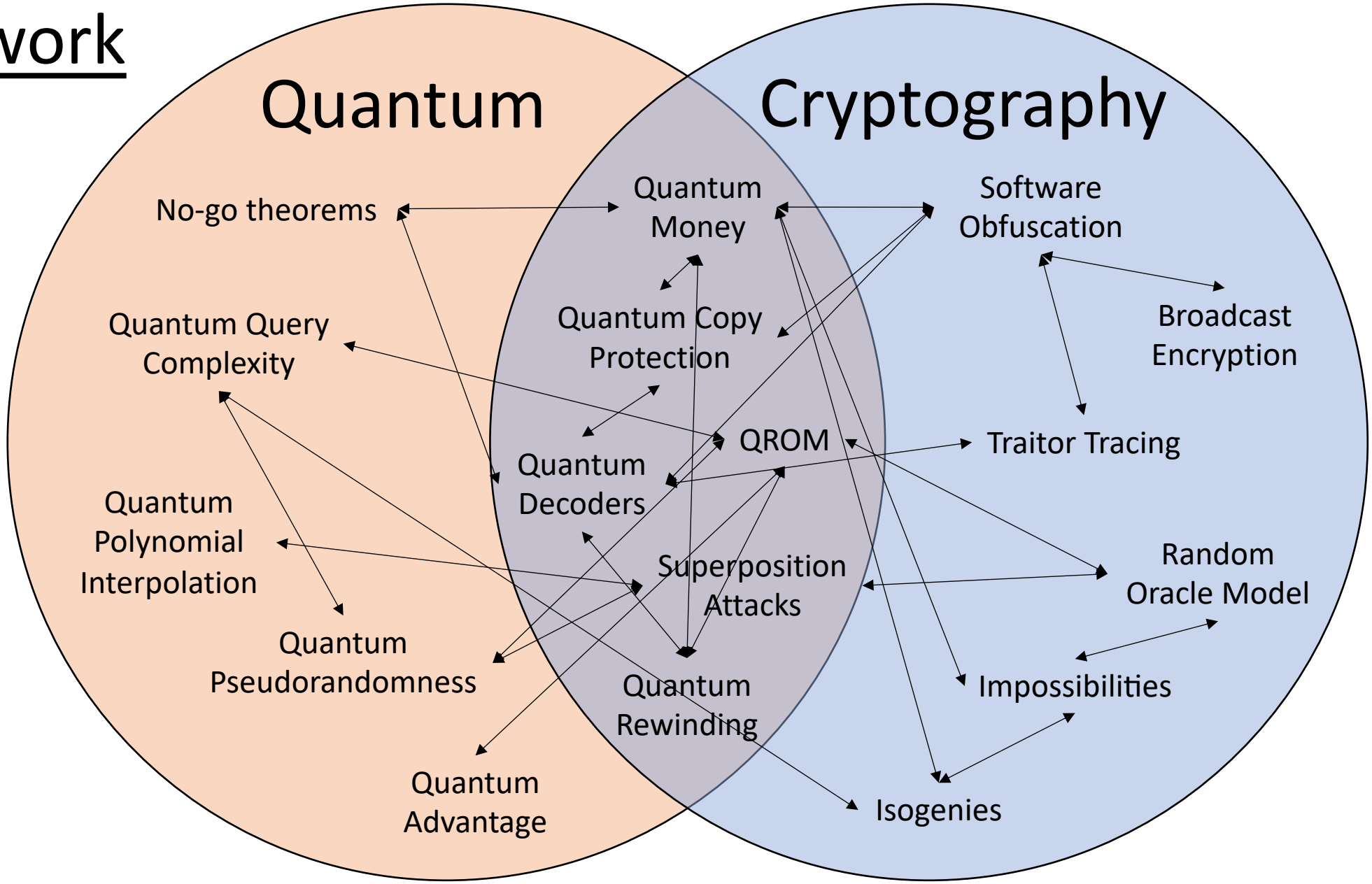


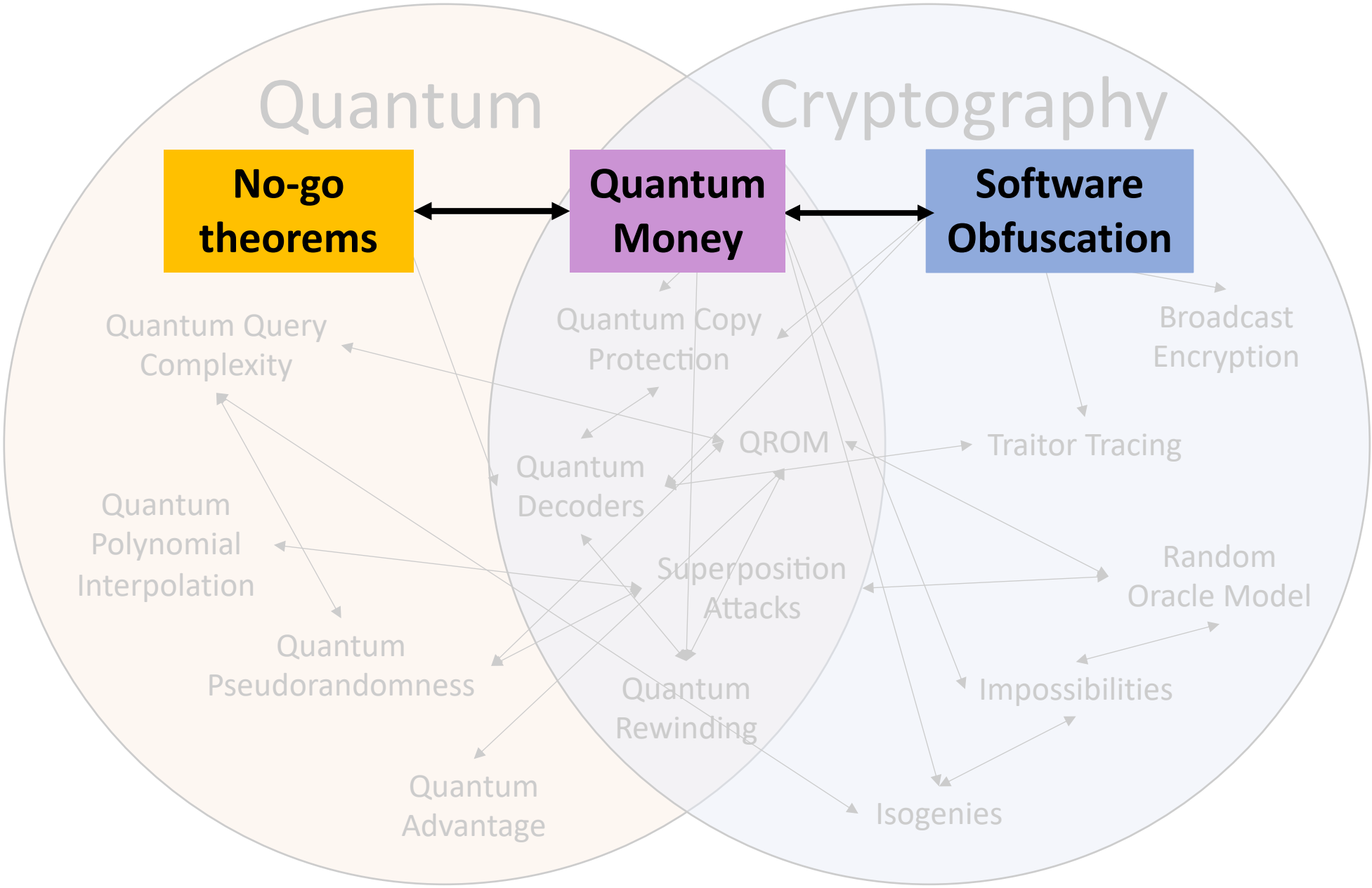
Quantum computers pose threats and new possibilities

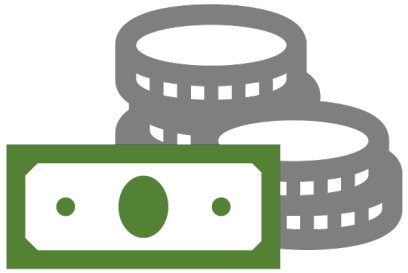
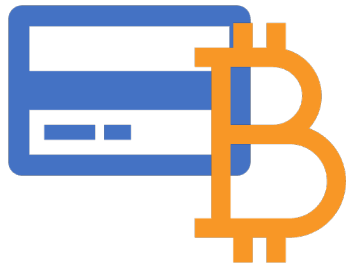
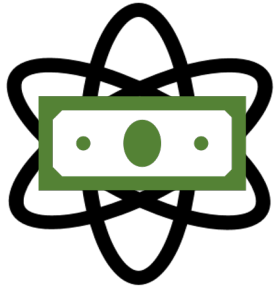
My work

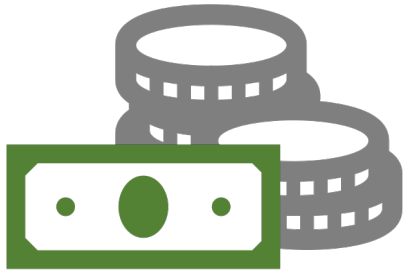
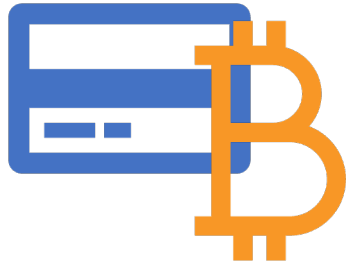
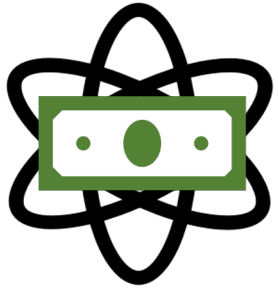

Quantum Computing
Understand the capabilities and limits of quantum computers
Quantum Cryptography
Protect against quantum computers; Harness quantum advantages
Cryptography
Provide solutions for novel cryptographic capabilities

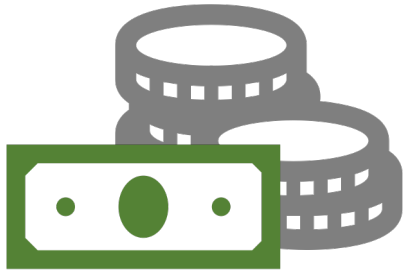
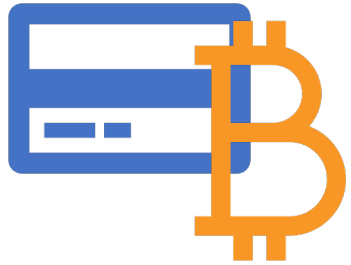
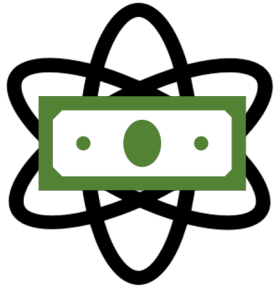
My work

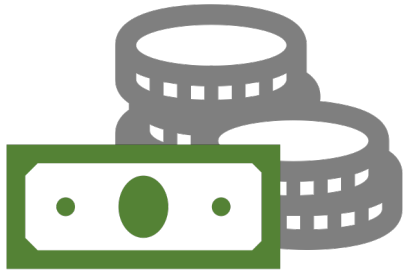
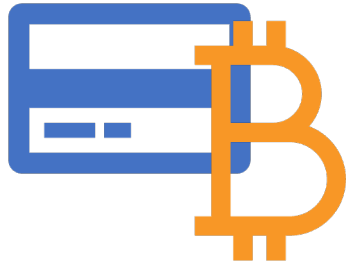
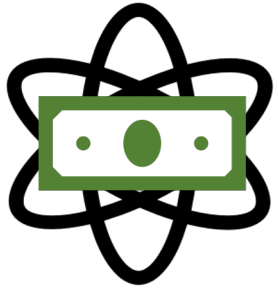

Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

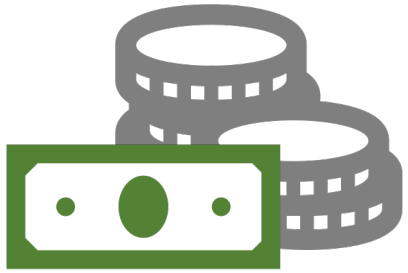
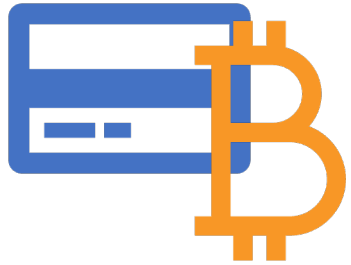
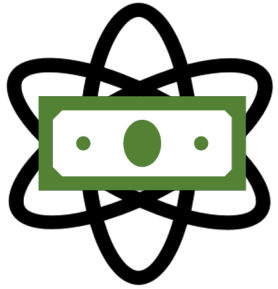
	Can send over internet?	Can verify ourselves?
Physical money		
Digital money		
(Public key) Quantum money		

	Can send over internet?	Can verify ourselves?
Physical money 	✗	✓
Digital money 		
(Public key) Quantum money 		

	Can send over internet?	Can verify ourselves?
Physical money 	✗	✓
Digital money 	✓	✗
(Public key) Quantum money 		

	Can send over internet?	Can verify ourselves?
Physical money 	✗	✓
Digital money 		
(Public key) Quantum money 		

Impossible to get both with classical computers:
can simply copy 1's and 0's

		Can send over internet?	Can verify ourselves?
Physical money		✗	✓
Digital money		✓	✗
(Public key) Quantum money		✓	✓

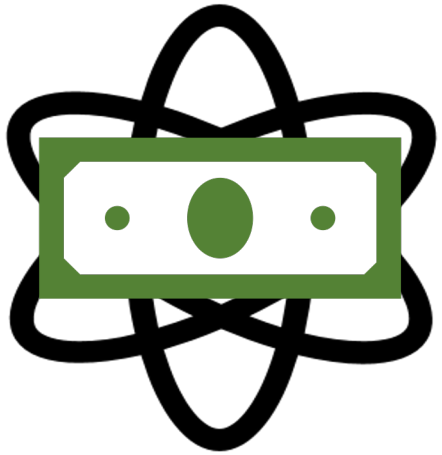
Quantum money is central to quantum cryptography

Quantum key distribution

[Bennet-Brassard'84, …]

Certified randomness

[Brakerski-Christiano-Mahadev-Vazirani-Vidick'18, **Z**'19a, Radian-Sattath'19, Yamakawa-**Z**'22, …]



Certified deletion

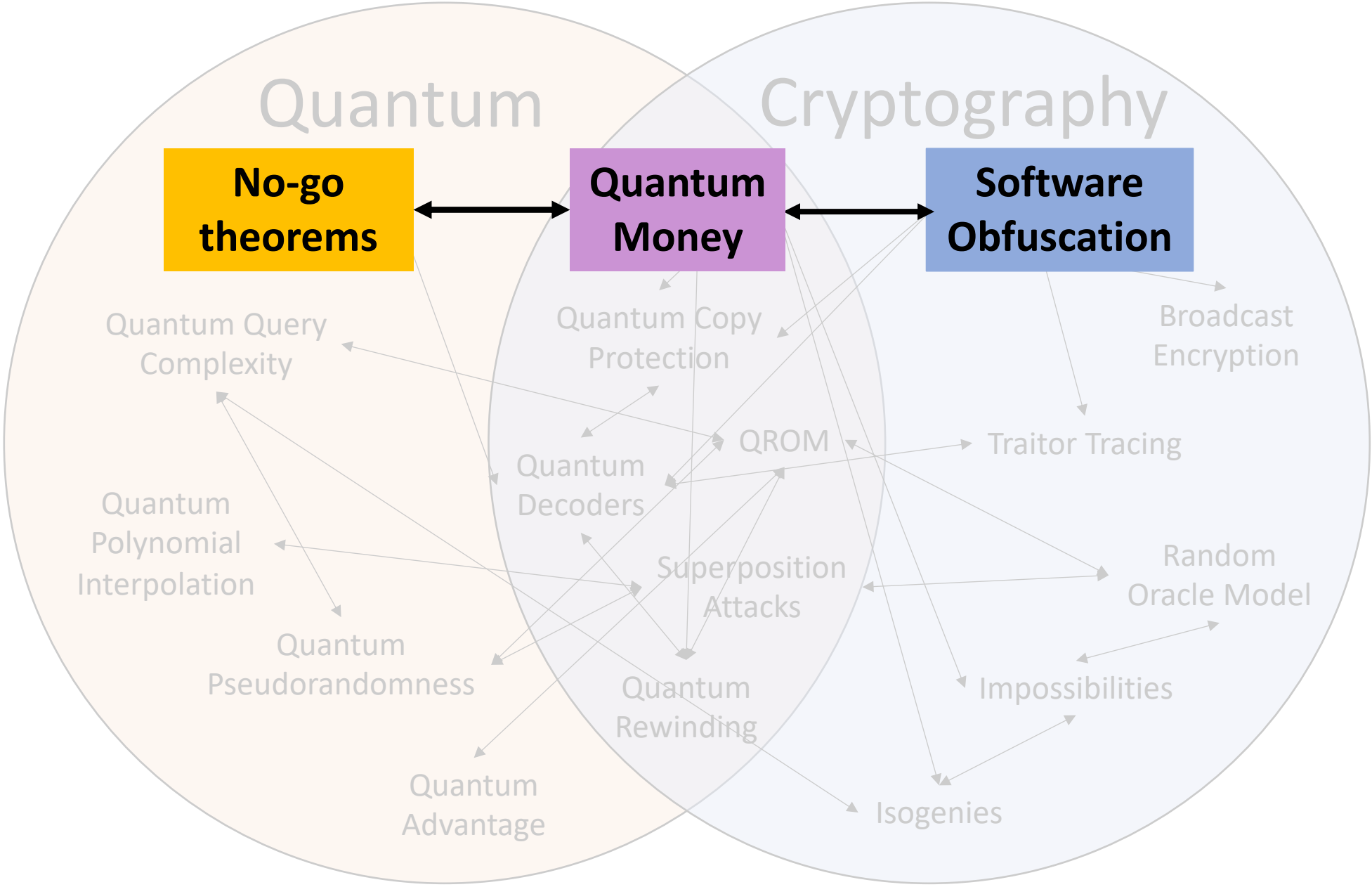
[Bartusek-Khurana'22, Poremba'23, Bartusek-Khurana-Malavolta-Poremba-Walter'23, Bartusek-Garg-Goyal-Khurana-Malavolta-Raizes-Roberts'23, …]

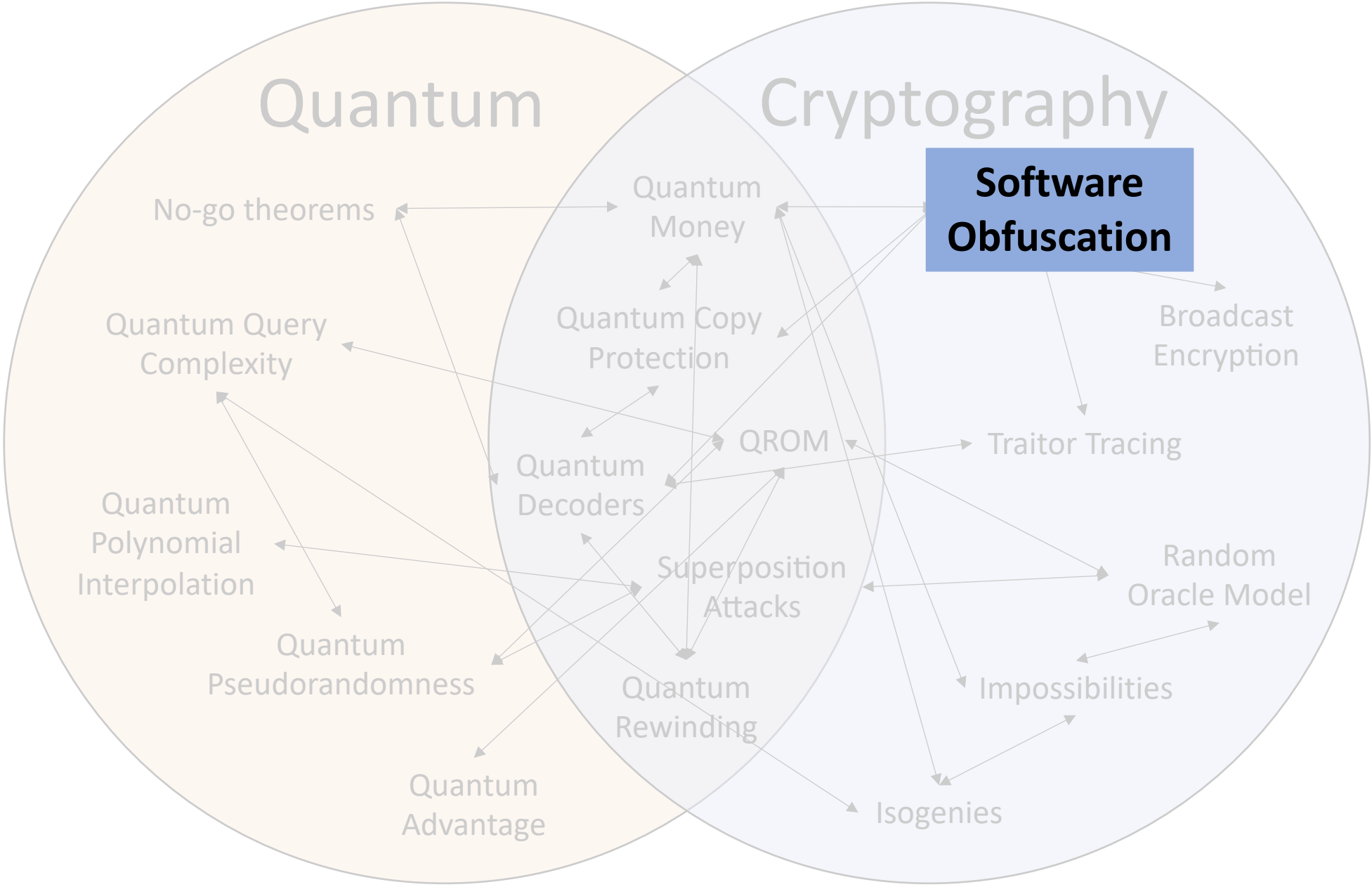
Quantum copy protection

[Aaronson'09, Ananth-La Placa'21, Broadbent-Jeffery-Lord-Podder-Sundaram'21, Kitagawa-Nishimaki-Yamakawa'21, Aaronson-Liu-Liu-**Z**-Zhang'21, Coladangelo-Liu-Liu-**Z**'21, Ananth-Kaleoglu-Li-Liu-**Z**'22 …]

Post-quantum secure hash functions, signatures

[**Z**'19a, Liu-**Z**'19, Liu-Montgomery-**Z**'23, **Z**'22, …]


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies


Quantum
Cryptography
Software Obfuscation
No-go theorems
Quantum Money
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

Program Obfuscation

Unobfuscated code

```
console.log("Hello World!");
```

Obfuscated code

```
function _0x3ee0(_0x4583f2,_0x55919d){var _0xdf5f12=_0xdf5f();return
_0x3ee0=function(_0x3ee016,_0x1f9768){_0x3ee016=_0x3ee016-0x149;var
_0x2d667c=_0xdf5f12[_0x3ee016];return
_0x2d667c;},_0x3ee0(_0x4583f2,_0x55919d);}var _0xf79053=_0x3ee0;function
_0xdf5f(){var
_0x3aa181=['24943570TbGodG','1703790ckzCVh','3780525fZcPAE','106821KYTjRw','3
08351BkJJmK','2RaolXo','Hello\x20World!','5810880saGYdH','272OJpmvb','116ysboPb'
,'log','116575sIfpxV'];_0xdf5f=function(){return _0x3aa181;};return
_0xdf5f();}(function(_0x422a42,_0x14cbd7){var
_0x43eb53=_0x3ee0,_0x126418=_0x422a42();while(!![]){try{var
_0x857bf4=parseInt(_0x43eb53(0x151))/0x1*(parseInt(_0x43eb53(0x152))/0x2)+-
parseInt(_0x43eb53(0x14e))/0x3+parseInt(_0x43eb53(0x14a))/0x4*(-
parseInt(_0x43eb53(0x14c))/0x5)+-
parseInt(_0x43eb53(0x154))/0x6+parseInt(_0x43eb53(0x14f))/0x7+-
parseInt(_0x43eb53(0x149))/0x8*(parseInt(_0x43eb53(0x150))/0x9)+parseInt(_0x43e
b53(0x14d))/0xa;if(_0x857bf4===_0x14cbd7)break;else
_0x126418['push'](_0x126418['shift']());}catch(_0x17622b){_0x126418['push'](_0x126
418['shift']());}}}(_0xdf5f,0xb16a4),console[_0xf79053(0x14b)](_0xf79053(0x153)));
```

Ad Hoc Obfuscation

Unobfuscated code

```
console.log("Hello World!");
```

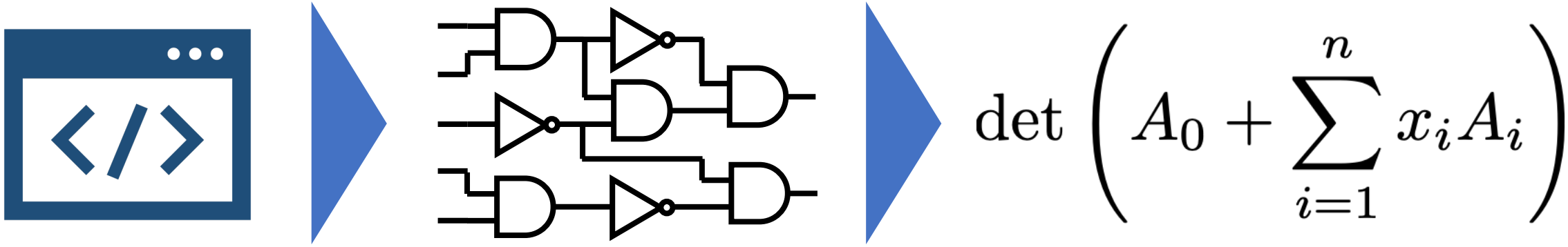
Obfuscated code

```
function _0x3ee0(_0x4583f2,_0x55919d){var _0xdf5f12=_0xdf5f();return
_0x3ee0=function(_0x3ee016,_0x1f9768){_0x3ee016=_0x3ee016-0x149;var
_0x2d667c=_0xdf5f12[_0x3ee016];return
_0x2d667c;},_0x3ee0(_0x4583f2,_0x55919d);}var _0xf79053=_0x3ee0;function
_0xdf5f(){var
_0x3aa181=['24943570TbGodG','1703790ckzCVh','3780525fZcPAE','106821KYTjRw','3
08351BkJJmK','2RaolXo','Hello\x20World!','5810880saGYdH','272OJpmvb','116ysboPb'
,'log','116575sIfpxV'];_0xdf5f=function(){return _0x3aa181;};return
_0xdf5f();}(function(_0x422a42,_0x14cbd7){var
_0x43eb53=_0x3ee0,_0x126418=_0x422a42();while(!![]){try{var
_0x857bf4=parseInt(_0x43eb53(0x151))/0x1*(parseInt(_0x43eb53(0x152))/0x2)+-
b53(0x14a))/0x4*(-
b53(0x14f))/0x7+-
50))/0x9)+parseInt(_0x43e
)break;else
_0x126418['push'](_0x126
4b)](_0xf79053(0x153)));
```

“Ad hoc” obfuscation:

- Code-level transformations
- No rigorous security guarantees
- Can typically be reverse engineered with sufficient effort

Cryptographic Program Obfuscation



First theorized by [Barak-Goldreich-Impagliazzo-Rudich-Sahai-Vadhan-Yang'01]
First candidate realization by [Garg-Gentry-Halevi-Raykova-Sahai-Waters'13]

Definition: Indistinguishability Obfuscation (iO)

1) Functionality preservation:

$$\mathrm{iO}(\boxed{\langle/\rangle})(x) = \boxed{\langle/\rangle}(x)$$

Definition: Indistinguishability Obfuscation (iO)

1) Functionality preservation:

$$iO(\langle/\rangle)(x) = \langle/\rangle(x)$$

2) Security:

Assume $\langle 1 \rangle$ and $\langle 2 \rangle$ are *functionally equivalent*

Definition: Indistinguishability Obfuscation (iO)

1) Functionality preservation:

$$iO(\langle/\rangle)(x) = \langle/\rangle(x)$$

2) Security:

Assume $\langle 1 \rangle$ and $\langle 2 \rangle$ are *functionally equivalent*

e.g. $\langle 1 \rangle(x, n) = \sum_{i=0}^{n-1} x^i$ $\quad$ $\langle 2 \rangle(x, n) = \frac{x^n - 1}{x - 1}$

Definition: Indistinguishability Obfuscation (iO)

1) Functionality preservation:

$$iO(\langle/\rangle)(x) = \langle/\rangle(x)$$

2) Security:

Assume $\langle 1 \rangle$ and $\langle 2 \rangle$ are *functionally equivalent*

$$\rightarrow iO(\langle 1 \rangle) \text{ vs } iO(\langle 2 \rangle)$$

My Work on iO

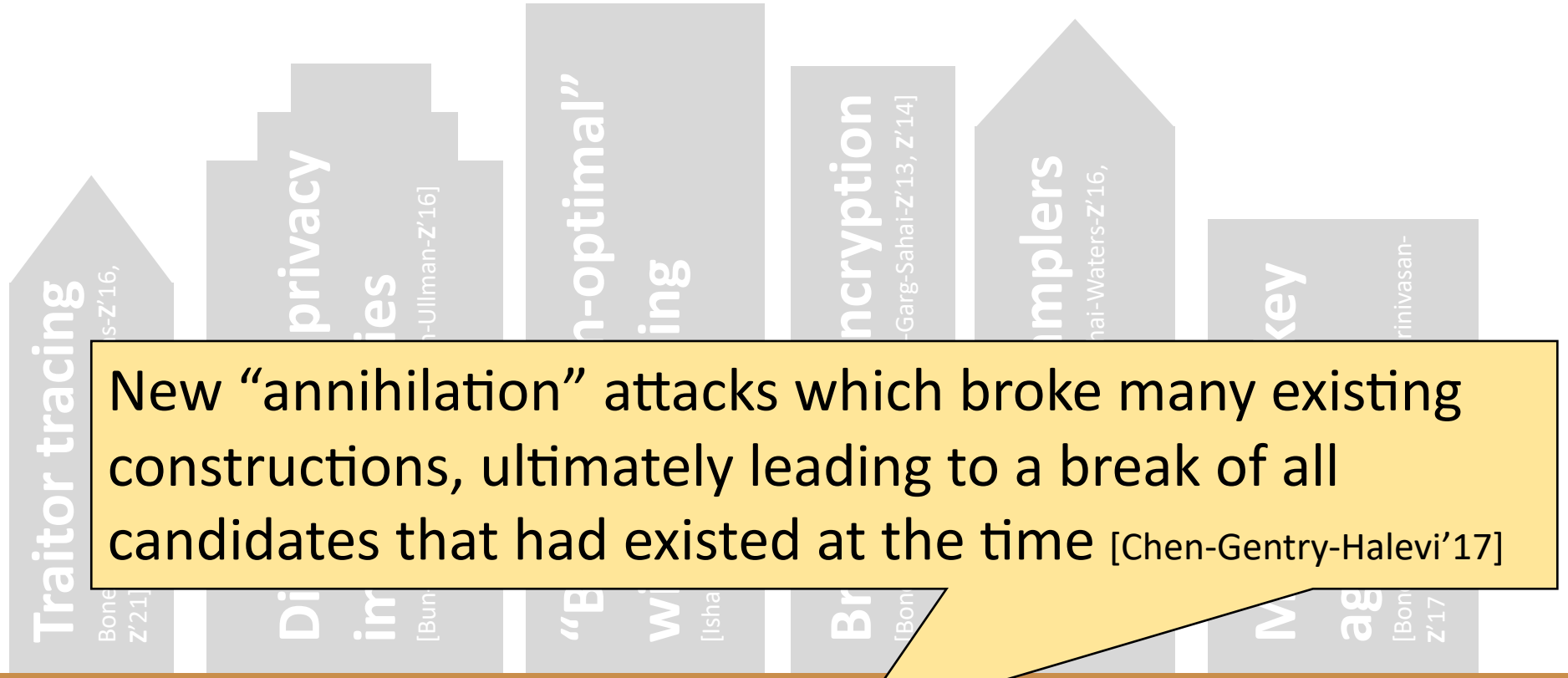
Applications

Vetting iO

[Badrinarayanan-Miles-Sahai-**Z**'16, Miles-Sahai-**Z**'16, Garg-Miles-Mukherjee-Sahai-Srinivasan-**Z**'16, Ma-**Z**'18, Bartusek-Guan-Ma-**Z**'18, Bartusek-Lepoint-Ma-**Z**'19, Bartusek-Ma-**Z**'19, Bartusek-Ishai-Jain-Ma-Sahai-**Z**'20]

My Work on iO

Applications



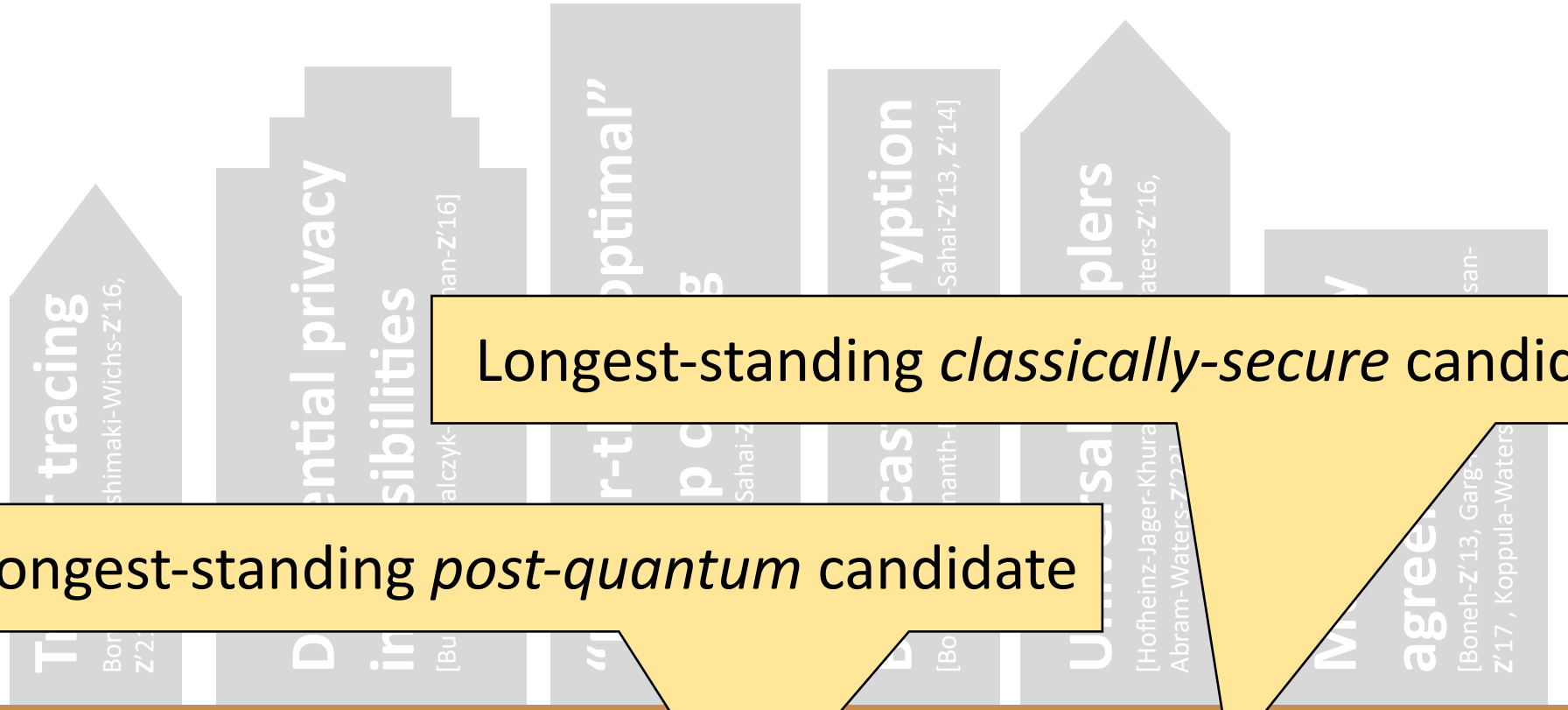
New "annihilation" attacks which broke many existing constructions, ultimately leading to a break of all candidates that had existed at the time [Chen-Gentry-Halevi'17]

Vetting iO

[Badrinarayanan-Miles-Sahai-**Z**'16, Miles-Sahai-**Z**'16, Garg-Miles-Mukherjee-Sahai-Srinivasan-**Z**'16, Ma-**Z**'18, Bartusek-Guan-Ma-**Z**'18, Bartusek-Lepoint-Ma-**Z**'19, Bartusek-Ma-**Z**'19, Bartusek-Ishai-Jain-Ma-Sahai-**Z**'20]

My Work on iO

Applications



Vetting iO

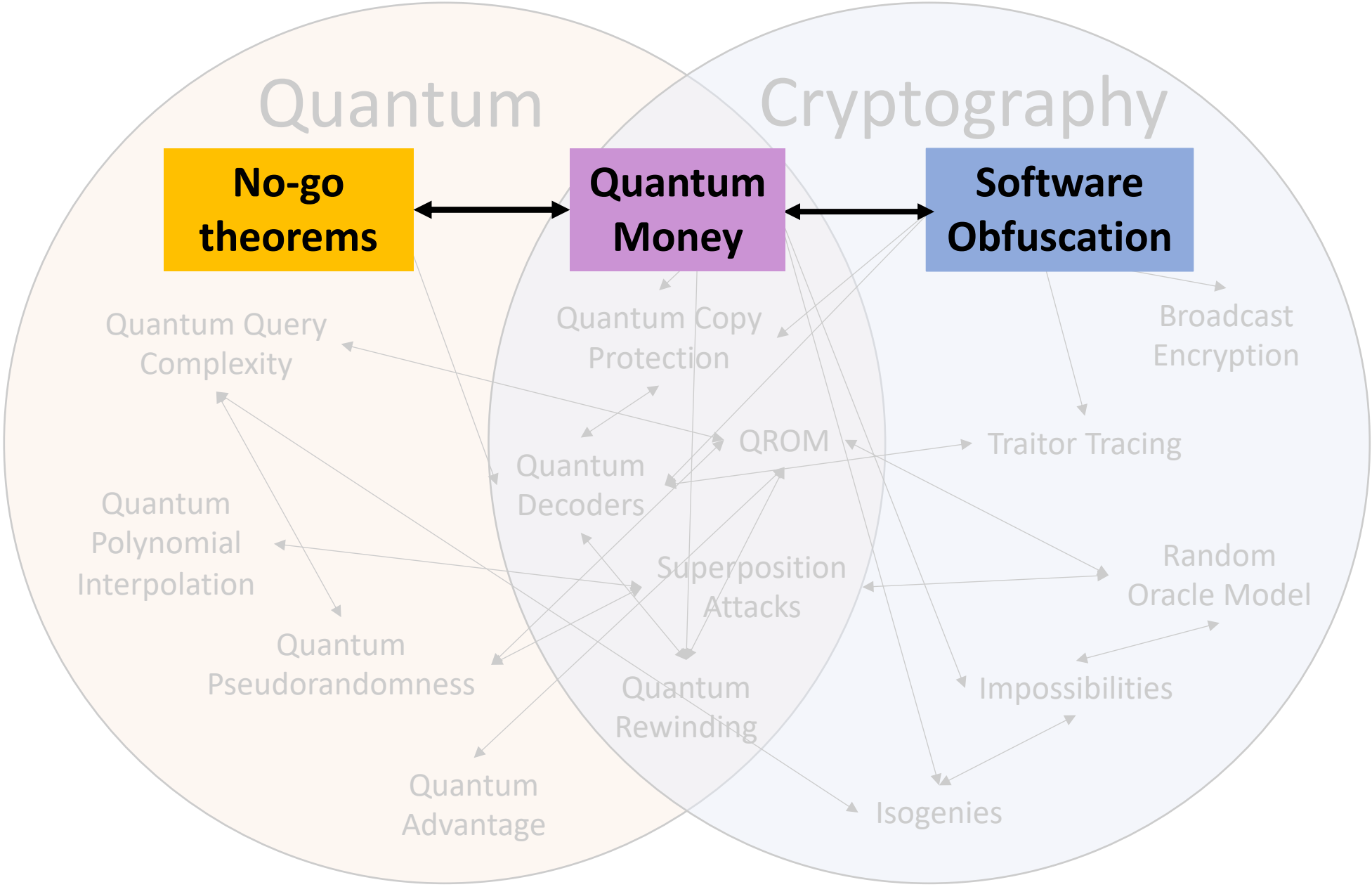
[Badrinarayanan-Miles-Sahai-**Z**'16, M[illegible]-Sahai-**Z**'16, Garg-Miles-Mukherjee-Sahai-Srinivasan-**Z**'16, Ma-**Z**'18, Bartusek-Guan-Ma-**Z**'18, Bartusek-Lepoint-Ma-**Z**'19, Bartusek-Ma-**Z**'19, Bartusek-Ishai-Jain-Ma-Sahai-**Z**'20]

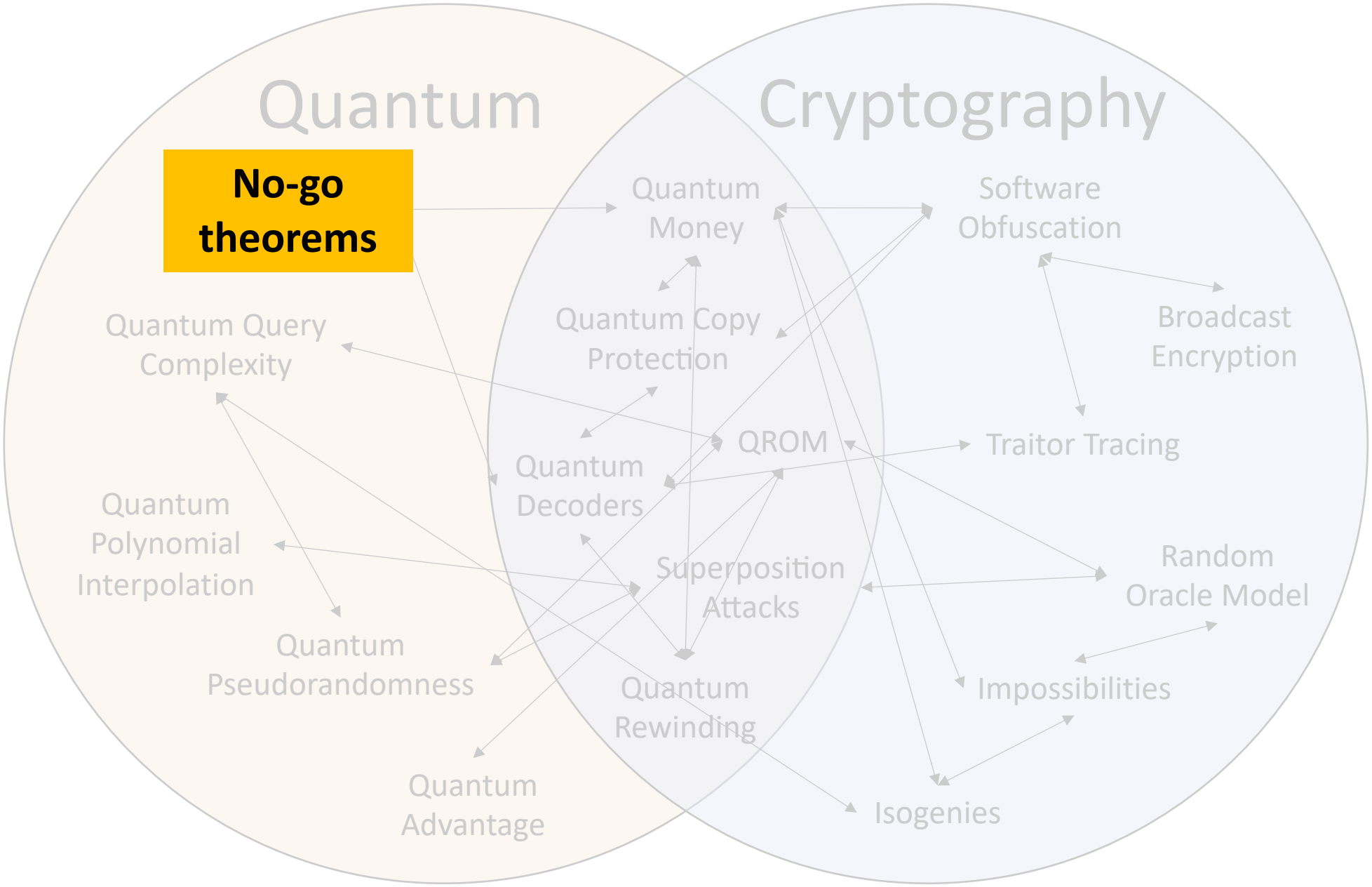
Open Questions – Future Work on iO

1. More efficient constructions

2. Better understanding of post-quantum security

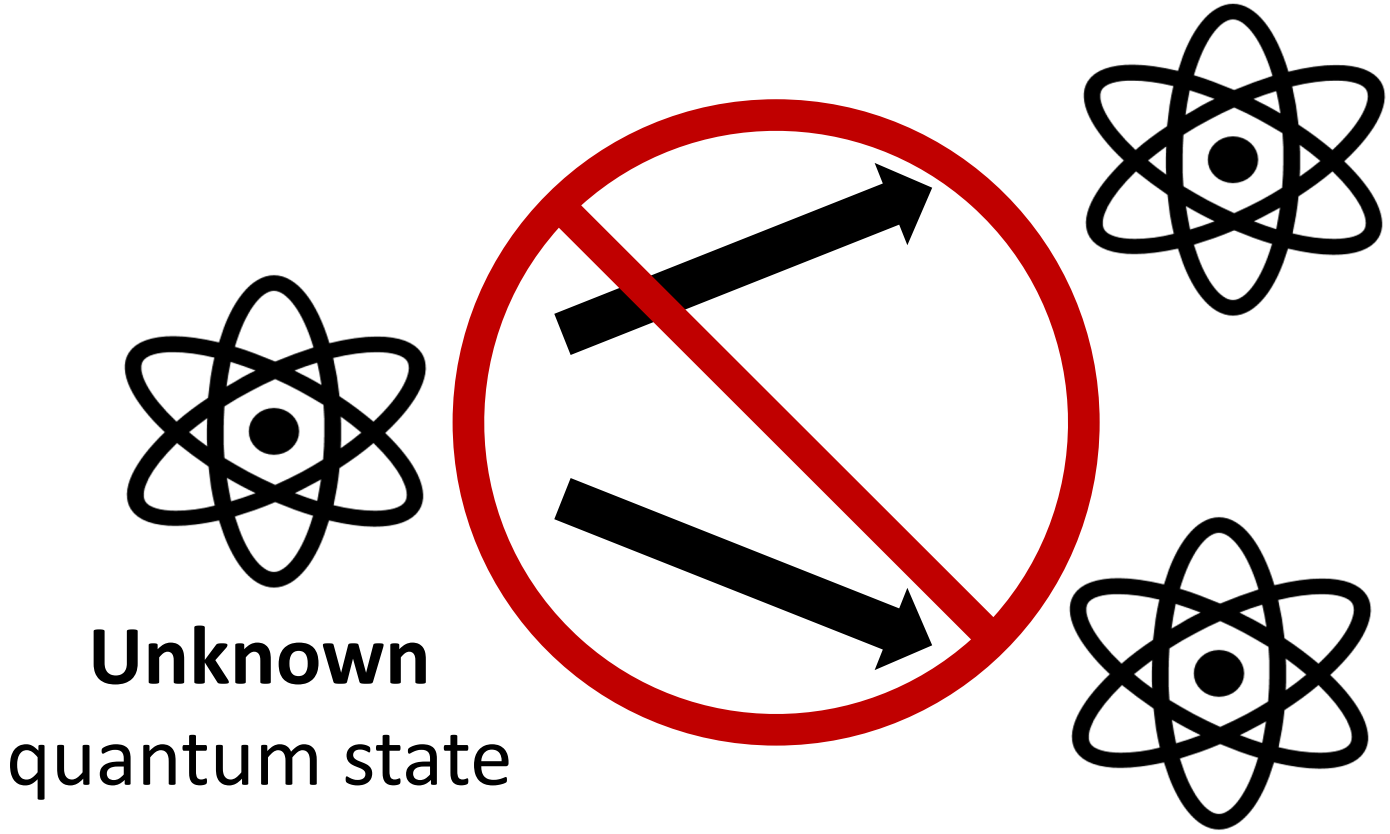
3. Obfuscation of quantum programs


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Copy Protection
Broadcast Encryption
Quantum Query Complexity
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

No-cloning Theorem

[Park'70, Wooters-Zurek'82, Dieks'82]

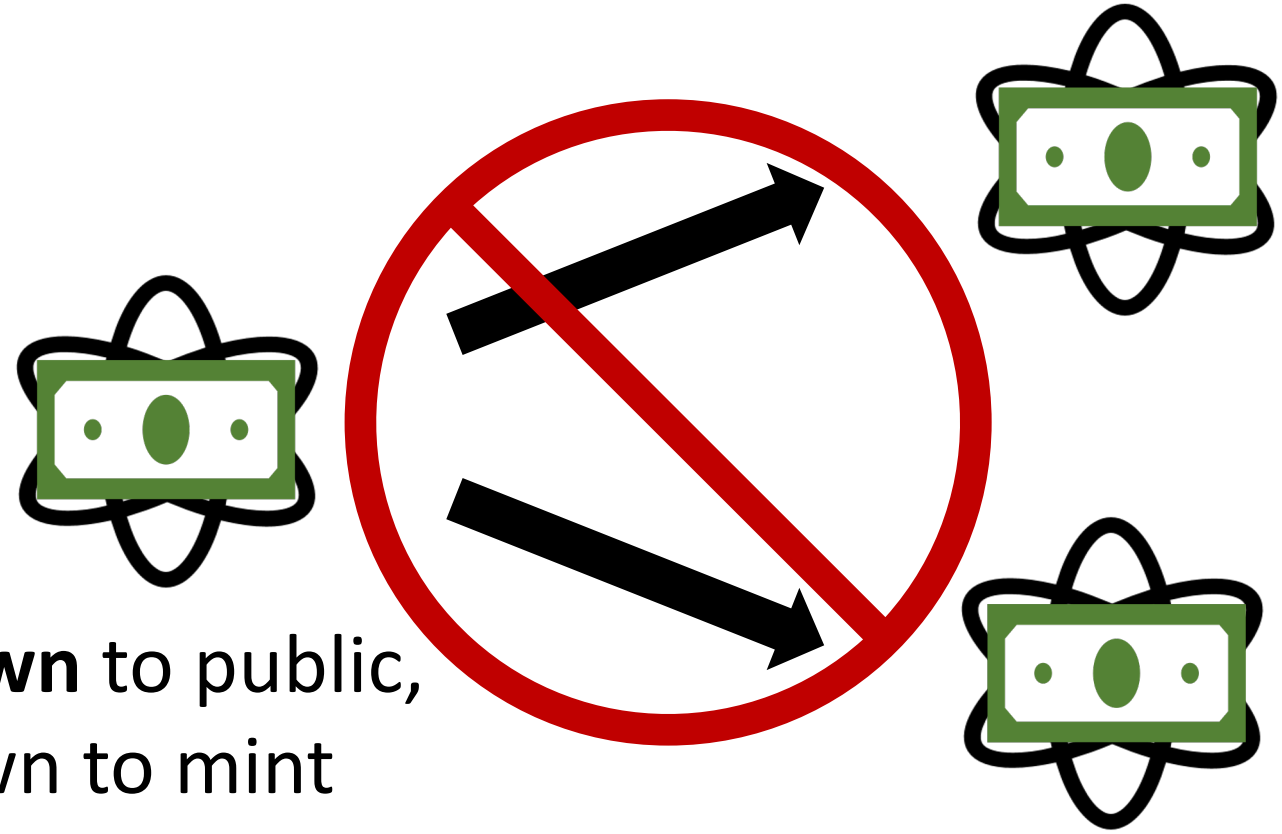


Unknown
quantum state

Other no-go theorems: No-teleporting, no-hiding, no-broadcast, no-deleting

Quantum Money

[Wiesner'70]



Unknown to public,
known to mint

Limitations of Wiesner's scheme

- **Public doesn't know state → mint needed to verify transactions**
- Certain attacks leveraging the mint's verification service

  [Lutomirski '10, Brodutch-Nagaj-Sattath-Unruh'14]
- Delicate quantum states decohere = self-destruct (Unknown state means they can't be fixed)

Limitations of Wiesner's scheme

- **Public doesn't know needed to verify transactions**
- Certain attacks leverag verification service

[Brodutch-Nagaj-Sattath-Unruh'14]

- Delicate quantum states d co elf-destruct (Unknown state means they can't be fixed)

Limitations of Wiesner's scheme

- **Public doesn't know needed to verify transactions**
- Certain attacks leverag verification service

  [Brodutch-Nagaj-Sattath-Unruh'14]
- Delicate quantum states d co elf-destruct (Unknown state means they can't be fixed)

***Public key* quantum money (PKQM)** = public can verify without mint

- First theorized by [Aaronson'09]
- Has since become a central goal in quantum cryptography

Information-Theory alone is not enough to construct Public-Key Quantum Money

Ability to verify → Banknotes are information-theoretically determined

→ (Information-theoretic) no-cloning does not apply

Central challenge: how to combine cryptographic and quantum information-theoretic techniques?

Attempts at constructing Public Key Quantum Money

= Security proof using widely studied tools

= Conjectured security

= Broken

[Aaronson'09]: random stabilizer states

✗ [Lutomirski-Aaronson-Farhi-Gosset-Hassidim-Kelner-Shor'10]

[Farhi-Gosset-Hassidim-Lutomirski-Shor'10]: knots

? [Liu-Montgomery-**Z**'23]

[Aaronson-Christiano'12]: polynomials hiding subspaces

✗ [Pena-Faugère-Perret'14, Christiano-Sattath'16]

[Kane'18, Kane-Sharif-Silverberg'21]: quaternion algebras

[**Z**'19a]: quadratic systems of equations

✗ [Roberts'21]

[Khesin-Lu-Shor'22]: lattices

✗ [Liu-Montgomery-**Z**'23]

Attempts at constructing Public Key Quantum Money

= Security proof using widely studied tools

= Conjectured security

= Broken

[Aaronson'09]: random stabilizer states

✗ [Lutomirski-Aaronson-Farhi-Gosset-Hassidim-Kelner-Shor'10]

[Farhi-Gosset-Hassidim-Lutomirski-Shor'10]: knots

? [Liu-Montgomery-**Z**'23]

[Aaronson-Christiano'12]: polynomials hiding subspaces

✗ [Pena-Faugère-Perret'14, Christiano-Sattath'16]

[Kane'18, Kane-Sharif-Silverberg'21]: quaternion algebras

[**Z**'19a]: quadratic systems of equations

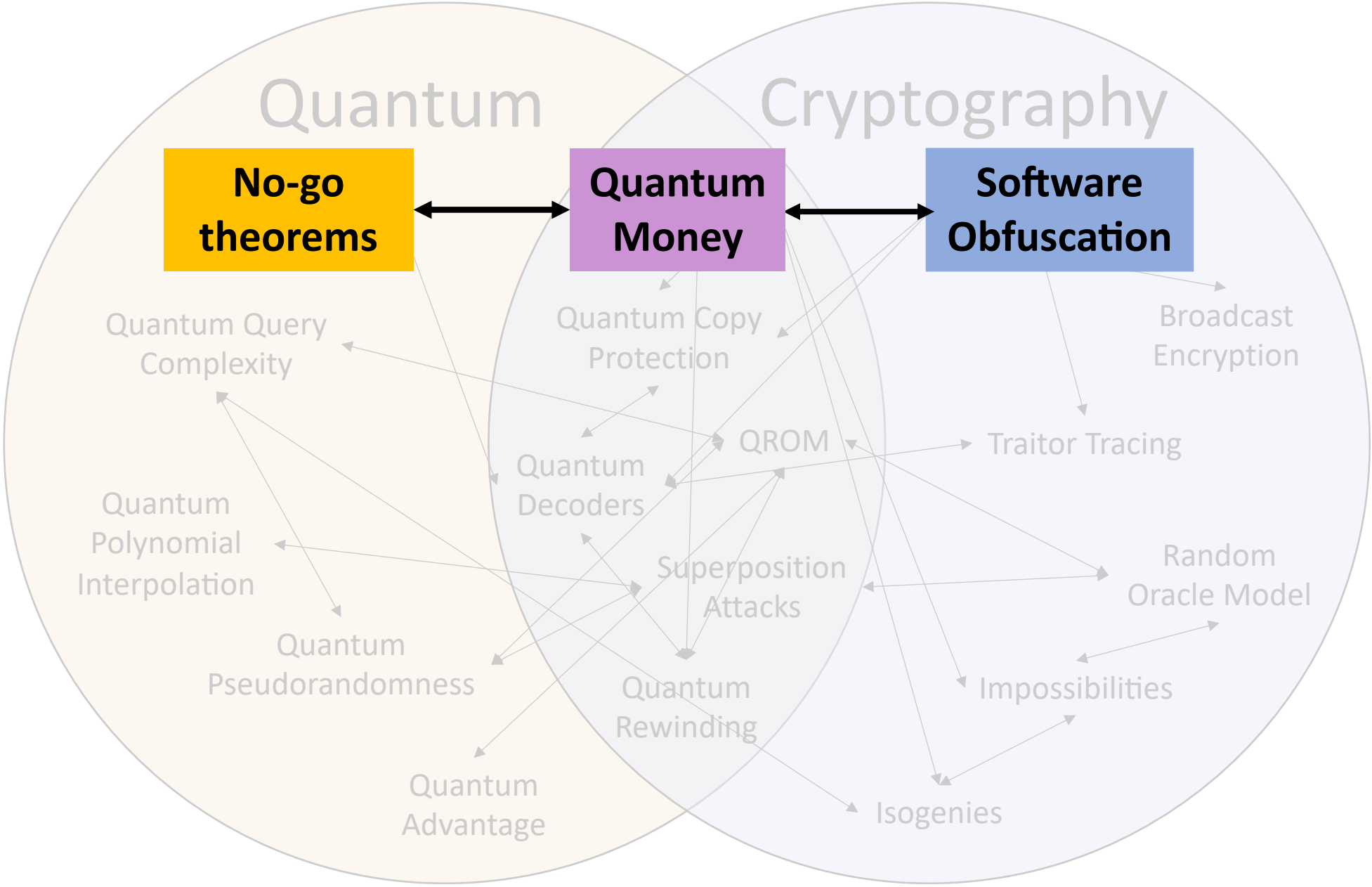
✗ [Roberts'21]

[Khesin-Lu-Shor'22]: lattices

✗ [Liu-Montgomery-**Z**'23]

[**Z**'19a]: program obfuscation

[**Z**'24]: group actions (isogenies over elliptic curves)


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Definition: Subspace hiding Obfuscation (shO)

$$P_S(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in S \\ 0 & \text{otherwise} \end{cases}$$

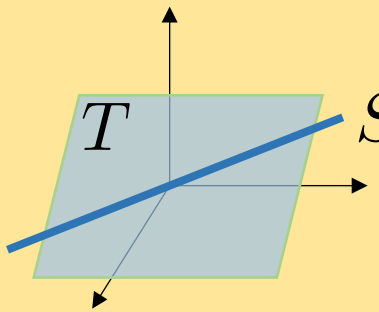
S a linear subspace

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Definition: Subspace hiding Obfuscation (shO)

$$P_S(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in S \\ 0 & \text{otherwise} \end{cases}$$

S a linear subspace



$S \subseteq T$

$\dim(T) \leq 3n/4$

T unknown

S known

$\text{shO}(P_S)$ vs $\text{shO}(P_T)$

S →

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Set $\mathsf{shO}(P_S) = \mathsf{iO}(P_S)$

Need to show

$$\mathsf{iO}\left(P_S = \begin{cases} 1 & \text{if } \mathbf{x} \in S \\ 0 & \text{otherwise} \end{cases}\right) \approx \mathsf{iO}\left(P_T = \begin{cases} 1 & \text{if } \mathbf{x} \in T \\ 0 & \text{otherwise} \end{cases}\right)$$

for known S but unknown T

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Set shO(

Recall: iO only guarantees anything for programs that are functionally identical.
Here $P_S(\mathbf{x}) \neq P_T(\mathbf{x})$ for $\mathbf{x} \in T \setminus S$

Need to show

$$\text{iO}\left(P_S = \begin{cases} 1 & \text{if } \mathbf{x} \in S \\ 0 & \text{otherwise} \end{cases}\right) \approx \text{iO}\left(P_T = \begin{cases} 1 & \text{if } \mathbf{x} \in T \\ 0 & \text{otherwise} \end{cases}\right)$$

for known S but unknown T

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Let G be a one-to-one function (will add more properties later)

Define a new function $Q_{\mathbf{A},y}(\mathbf{x})$:

Let $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

Output $\begin{cases} 1 & \text{if } \mathbf{z} = \mathbf{0} \text{ or } \mathsf{G}(\mathbf{z}/|\mathbf{z}|) = y \\ 0 & \text{otherwise} \end{cases}$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Let $\mathbf{A}$ be a matrix such that $\mathbf{x} \in S \iff \mathbf{z} = \mathbf{0}$

where $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

Let y_0 be some *non-image* of G

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Let $\mathbf{A}$ be a matrix such that $\mathbf{x} \in S \iff \mathbf{z} = \mathbf{0}$

where $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

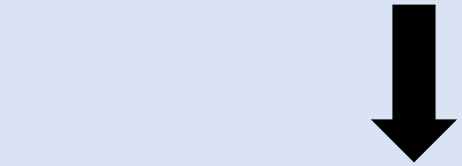
Let y_0 be some *non-image* of G

$$Q_{\mathbf{A},y_0}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{z} = \mathbf{0} \text{ or } \cancel{\mathsf{G}(\mathbf{z}/|\mathbf{z}|) = y_0} \\ 0 & \text{otherwise} \end{cases}$$

$$\Rightarrow \boxed{Q_{\mathbf{A},y_0}(\mathbf{x}) = P_S(\mathbf{x})}$$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$$Q_{\mathbf{A},y_0}(\mathbf{x}) = P_S(\mathbf{x})$$



iO security

$$\mathsf{iO}(P_S) \approx \mathsf{iO}(Q_{\mathbf{A},y_0})$$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Assume that $\dim(T) = \dim(S) + 1$ (can extend to general setting by repetition)

➡ $\exists \mathbf{v} : \mathbf{x} \in T \iff \mathbf{z} \in \mathrm{Span}(\mathbf{v})$ where $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

Let $y_1 = \mathsf{G}(\mathbf{v}/|\mathbf{v}|)$

↑

unknown

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$$\mathsf{iO}(P_S) \approx \mathsf{iO}(Q_{\mathbf{A},y_0}) \qquad \mathsf{iO}(Q_{\mathbf{A},y_1})$$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$$\mathsf{iO}(P_S) \approx \mathsf{iO}(Q_{\mathbf{A}}, y_0) \approx \mathsf{iO}(Q_{\mathbf{A}}, y_1)$$

Non-image

Image on unknown point

Definition (informal): G is a pseudorandom generator if images on unknown points are indistinguishable from non-images

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Recall $Q_{\mathbf{A},y}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{z} = \mathbf{0} \text{ or } \mathsf{G}(\mathbf{z}/|\mathbf{z}|) = y \\ 0 & \text{otherwise} \end{cases}$

where $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

Observe

$$\mathbf{x} \in T \iff \mathbf{z} \in \mathsf{Span}(\mathbf{v})$$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Recall $Q_{\mathbf{A},y}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{z} = \mathbf{0} \text{ or } \mathsf{G}(\mathbf{z}/|\mathbf{z}|) = y \\ 0 & \text{otherwise} \end{cases}$

where $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

Observe

$$\mathbf{x} \in T \iff \mathbf{z} = \mathbf{0} \text{ or } \frac{\mathbf{z}}{|\mathbf{z}|} = \frac{\mathbf{v}}{|\mathbf{v}|}$$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + Lem: shO ⇒ PKQM

Recall $Q_{\mathbf{A},y}(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{z} = \mathbf{0} \text{ or } \mathsf{G}(\mathbf{z}/|\mathbf{z}|) = y \\ 0 & \text{otherwise} \end{cases}$

where $\mathbf{z} = \mathbf{A} \cdot \mathbf{x}$

Observe

$$\mathbf{x} \in T \iff \mathbf{z} = \mathbf{0} \text{ or } \mathsf{G}\left(\frac{\mathbf{z}}{|\mathbf{z}|}\right) = \mathsf{G}\left(\frac{\mathbf{v}}{|\mathbf{v}|}\right)$$

y_1

$$Q_{\mathbf{A},y_1}(\mathbf{x}) = P_T(\mathbf{x})$$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$$\mathsf{iO}(P_S) \approx \mathsf{iO}(Q_{\mathbf{A},y_0}) \approx \mathsf{iO}(Q_{\mathbf{A},y_1}) \approx \mathsf{iO}(P_T)$$

↑ iO

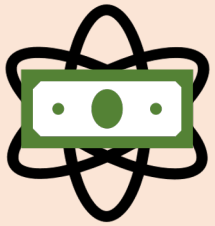
↑ Pseudorandom generator

↑ iO

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

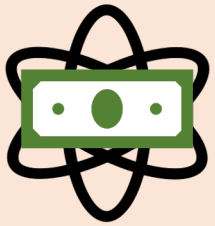
Follow blueprint of [Aaronson-Christiano'12]



$$= |\$_S\rangle := \frac{1}{\sqrt{|S|}} \sum_{\mathbf{x} \in S} |\mathbf{x}\rangle \qquad S = \text{secret subspace of dimension } n/2$$

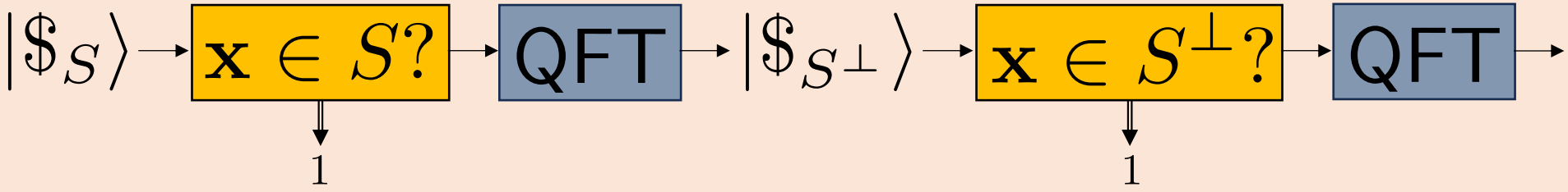
Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Follow blueprint of [Aaronson-Christiano'12]



$$= |\$_S\rangle := \frac{1}{\sqrt{|S|}} \sum_{\mathbf{x} \in S} |\mathbf{x}\rangle \qquad S = \text{secret subspace of dimension } n/2$$

Verification:

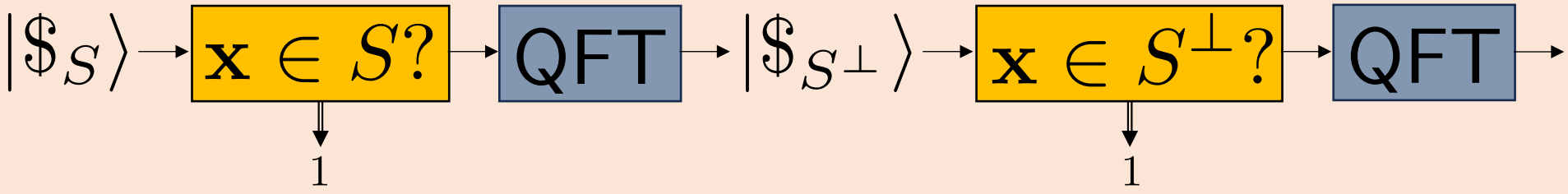


Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Follow blueprint of [Aaronson-Christiano'12]

My work: implement using $\mathsf{shO}(S)$ and $\mathsf{shO}(S^\perp)$

Verification:

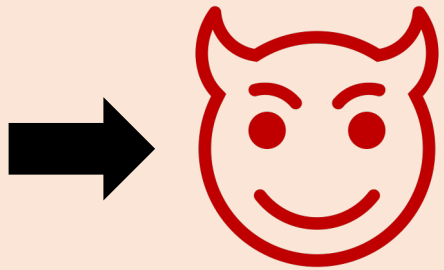


Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$|\$_S\rangle$

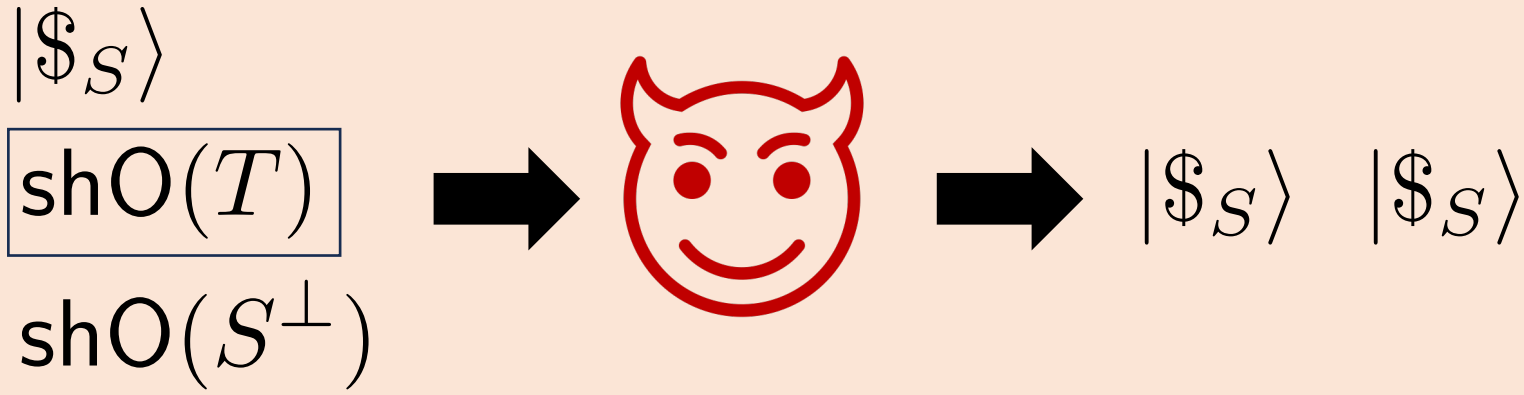
$\mathsf{shO}(S)$

$\mathsf{shO}(S^\perp)$



$|\$_S\rangle \quad |\$_S\rangle$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

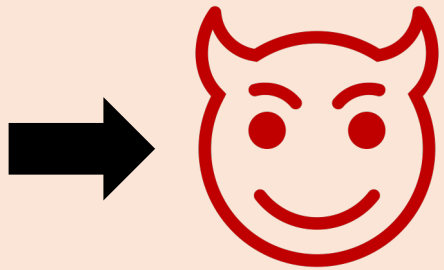


Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$|\$_S\rangle$

$\mathsf{shO}(T)$

$\boxed{\mathsf{shO}(U)}$



$|\$_S\rangle \quad |\$_S\rangle$

$U^\perp \subseteq S \subseteq T$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

$|\$_S\rangle$

$\mathsf{shO}(T)$

$\mathsf{shO}(U)$

$|\$_S\rangle \quad |\$_S\rangle$

$U^\perp \subseteq S \subseteq T$

Quantitative No-Cloning Theorem:

$T\ U\ |\$_S\rangle$ ↛ $|\$_S\rangle \quad |\$_S\rangle$

Thm: iO ⇒ PKQM = **Lem:** iO ⇒ shO + **Lem:** shO ⇒ PKQM

Surprising Applications of Quantum Money Techniques

Copy Protection

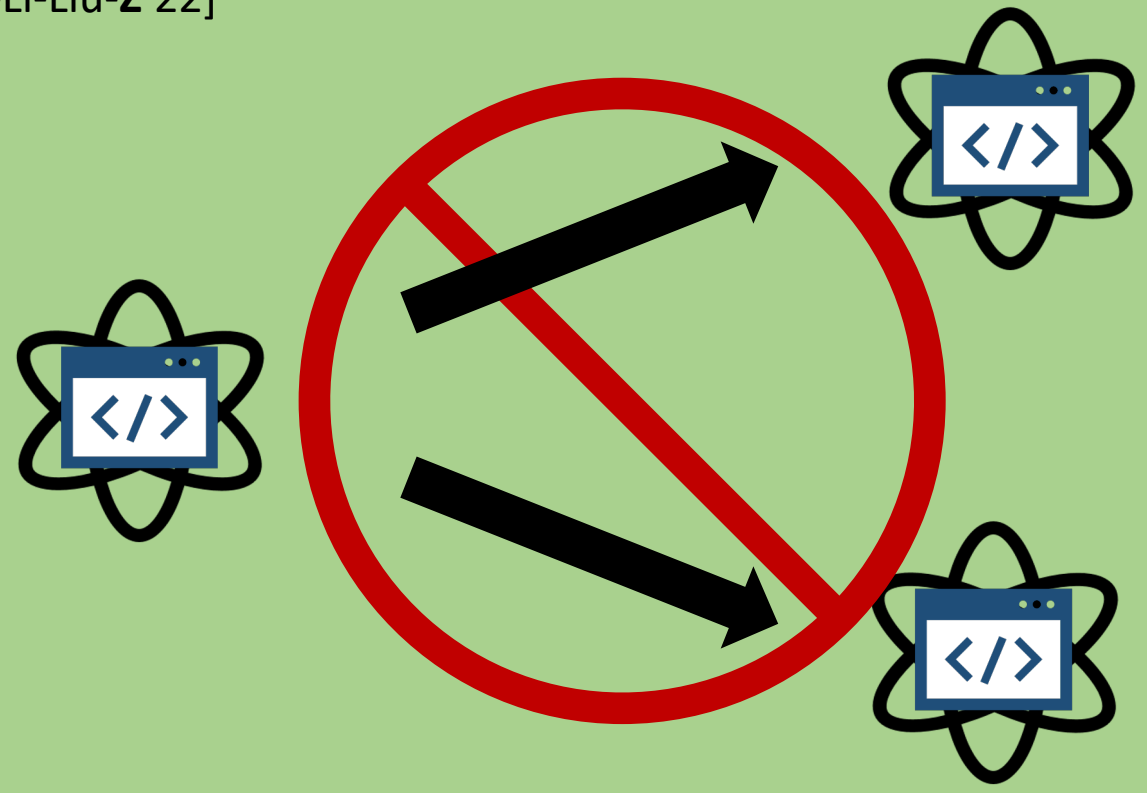
Quantum Lightning

One-shot Signatures

Surprising Applications of Quantum Money Techniques

Copy Protection

[Aaronson'09, Aaronson-Liu-Liu-**Z**-Zhang'21, Coladangelo-Liu-Liu-**Z**'21, Ananth-Kaleoglu-Li-Liu-**Z**'22]



Quantum Lightning

One-shot Signatures

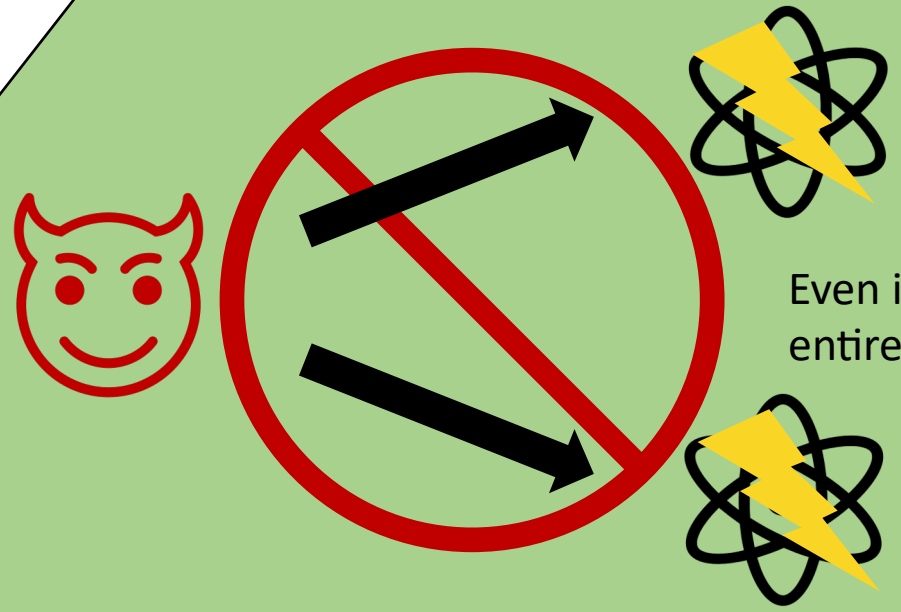
Surprising Applications of Quantum Money Techniques

Copy Protection

Quantum Lightning

One-shot Signatures

[Lutomirski-Aaronson-Farhi-Gosset-Hassidim-Kelner-Shor09, **Z**'19a, **Z**'24]



Even if adversary controlled the entire process to create the state

Decentralized quantum money
("Blockchain without the blockchain")

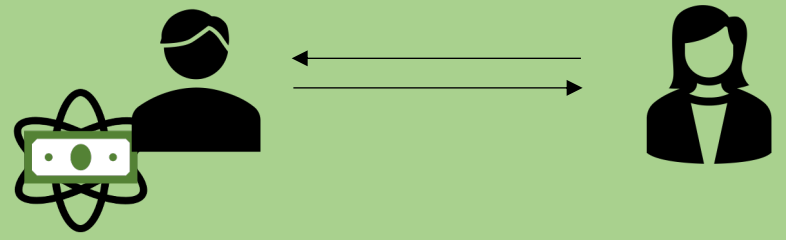
Surprising Applications of Quantum Money Techniques

Copy Protection

Quantum Lightning

One-shot Signatures

[Amos-Georgiou-Kiayias-**Z**'20]



Quantum money with classical communication

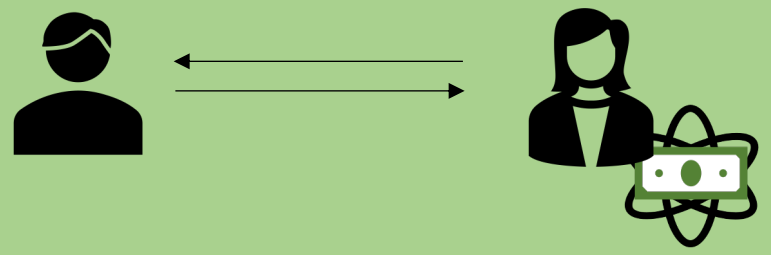
Surprising Applications of Quantum Money Techniques

Copy Protection

Quantum Lightning

One-shot Signatures

[Amos-Georgiou-Kiayias-**Z**'20]



Quantum money with classical communication

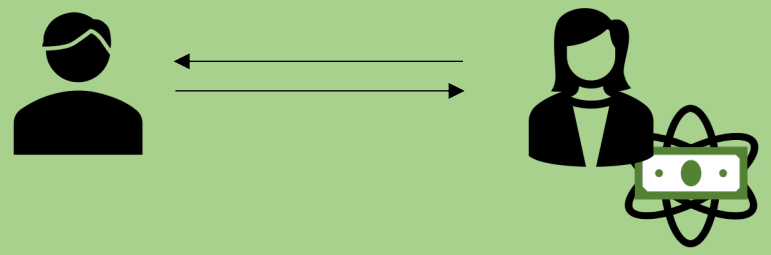
Surprising Applications of Quantum Money Techniques

Copy Protection

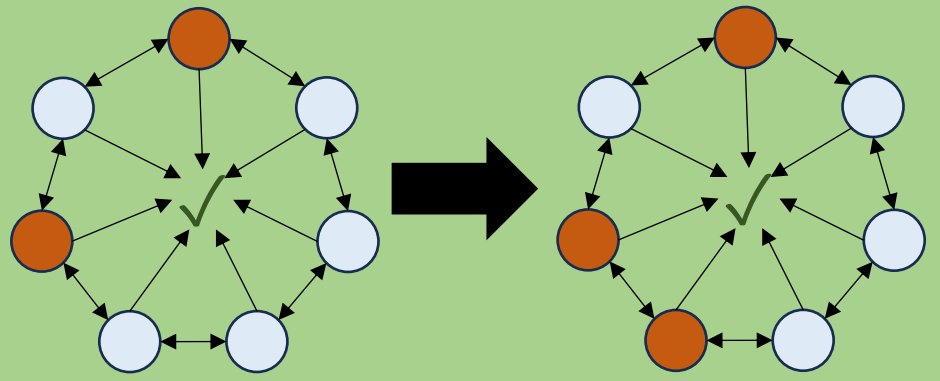
Quantum Lightning

One-shot Signatures

[Amos-Georgiou-Kiayias-**Z**'20]



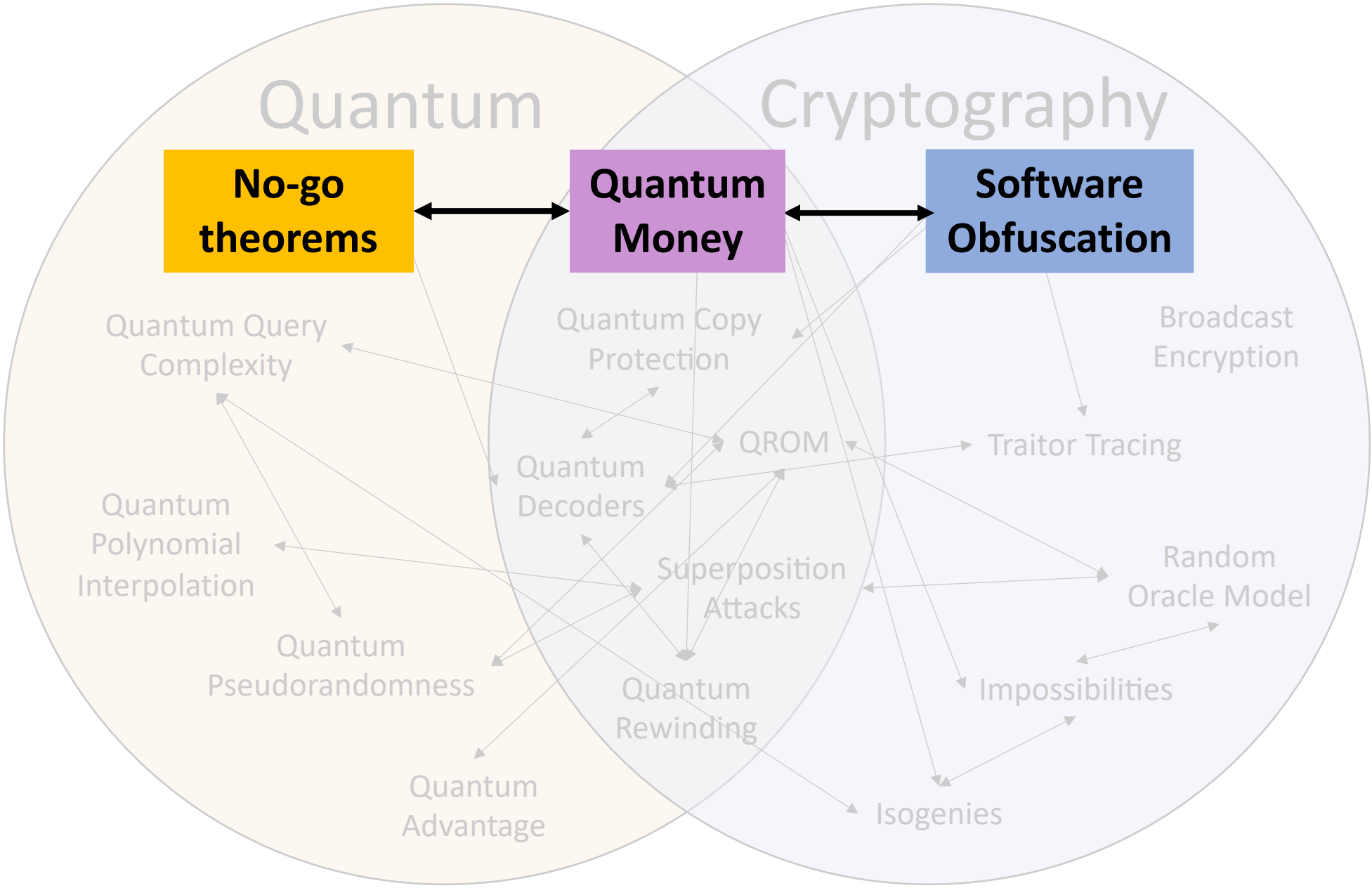
Quantum money with classical communication



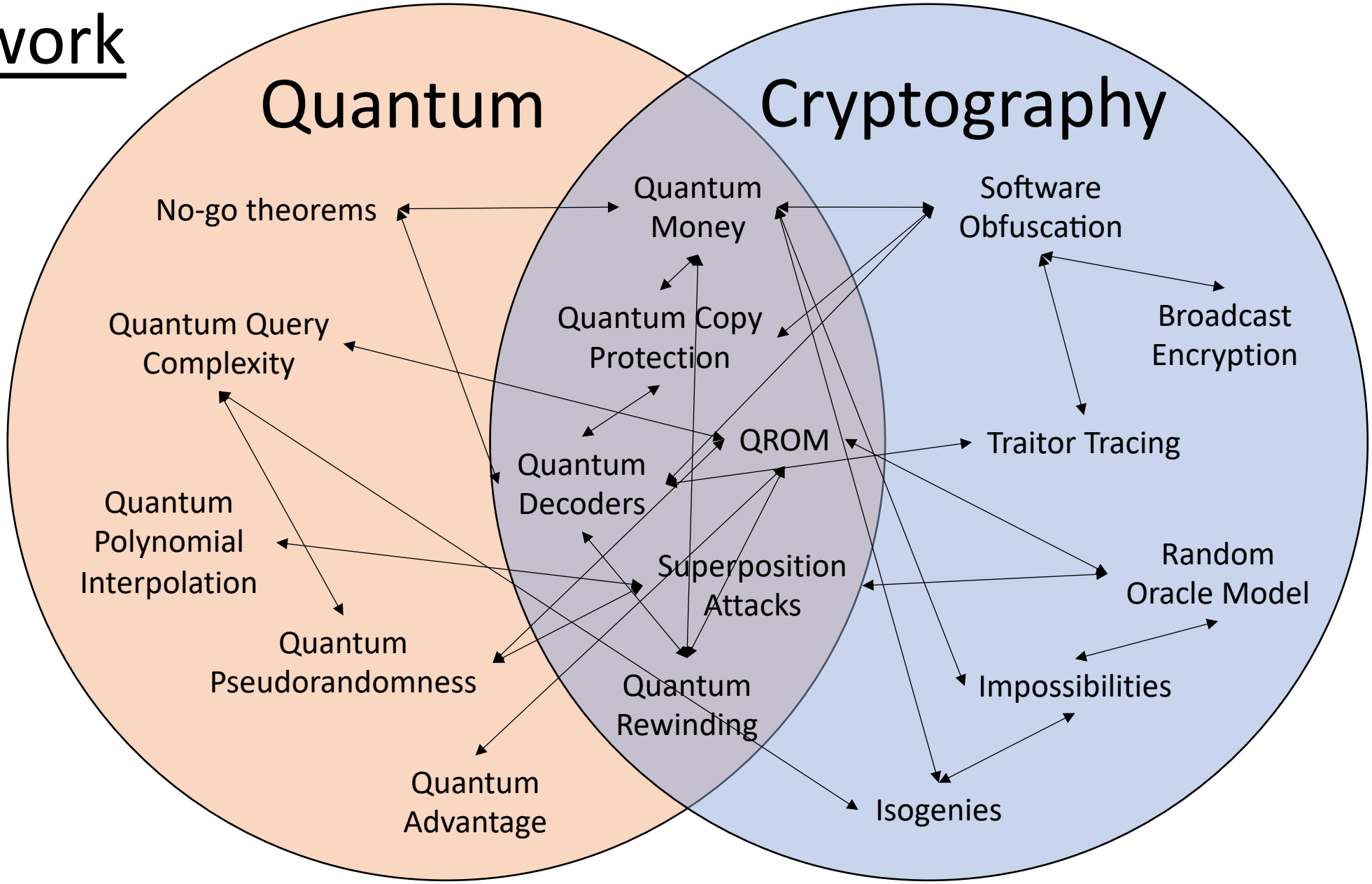
Overcoming consensus lower-bounds [Drake]

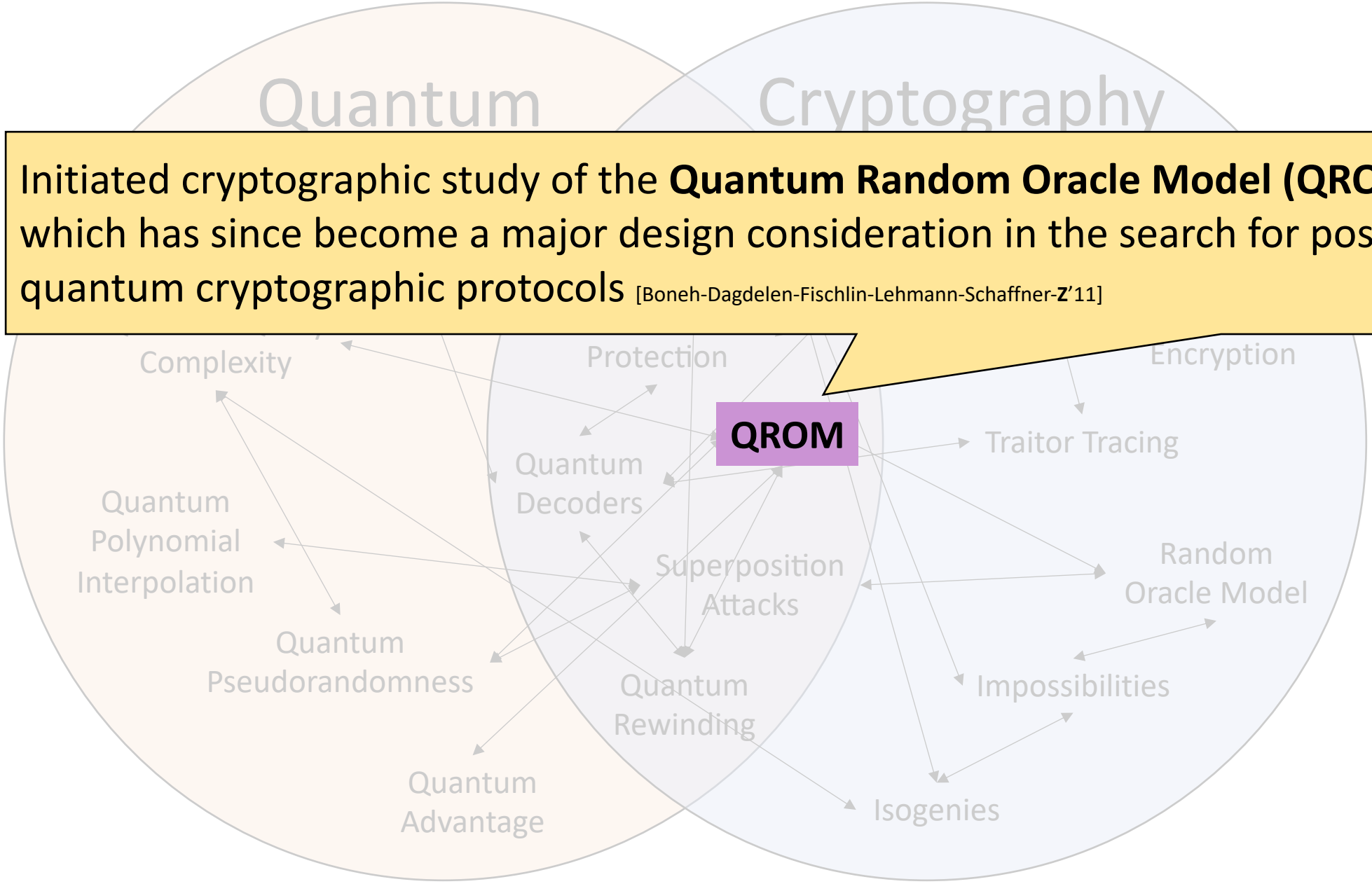
Open Questions – Future Work on Quantum Money

1. Build from other cryptographic tools
2. Other possible extensions
3. Quantum protocols beyond no-cloning

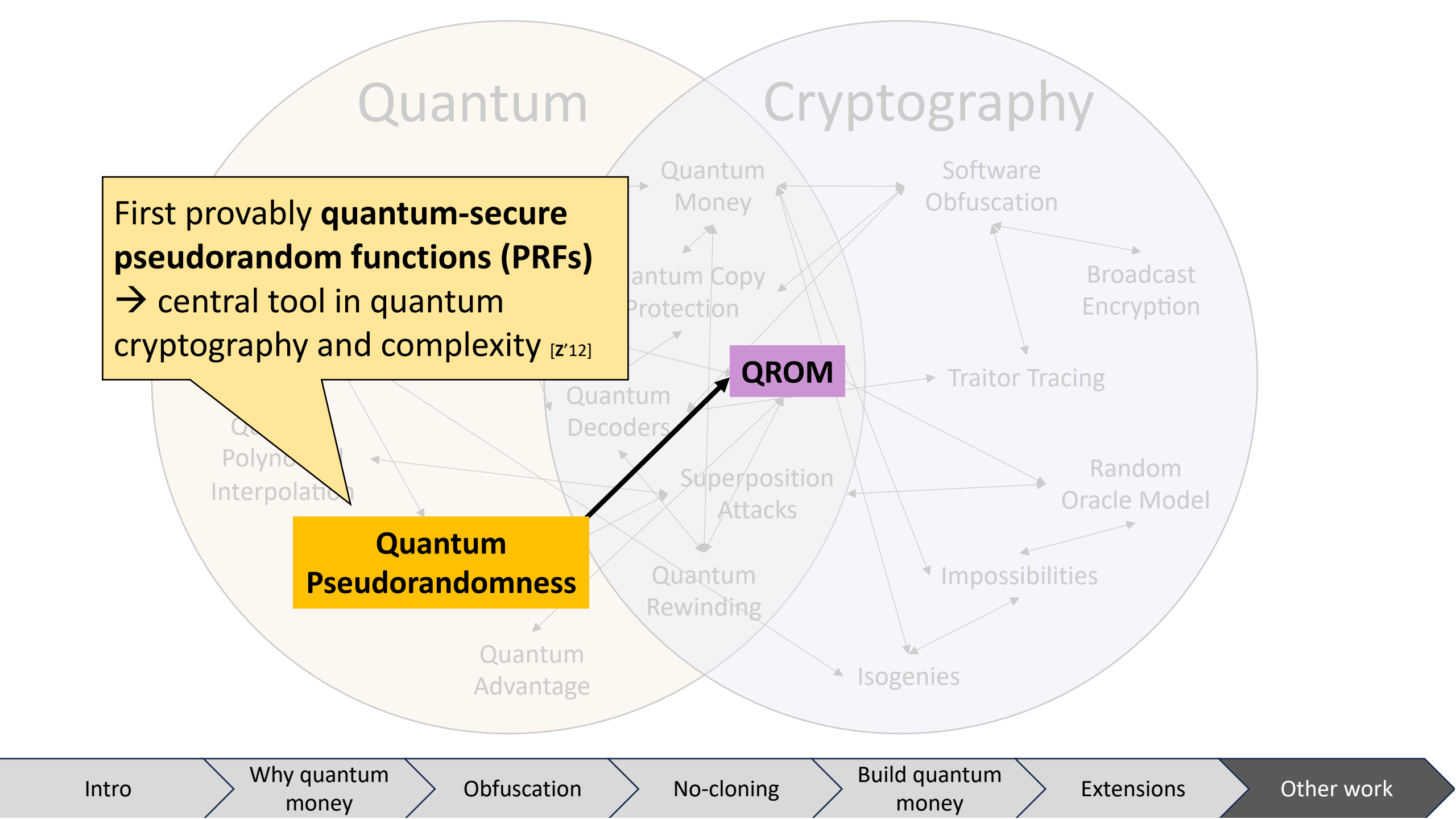

Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

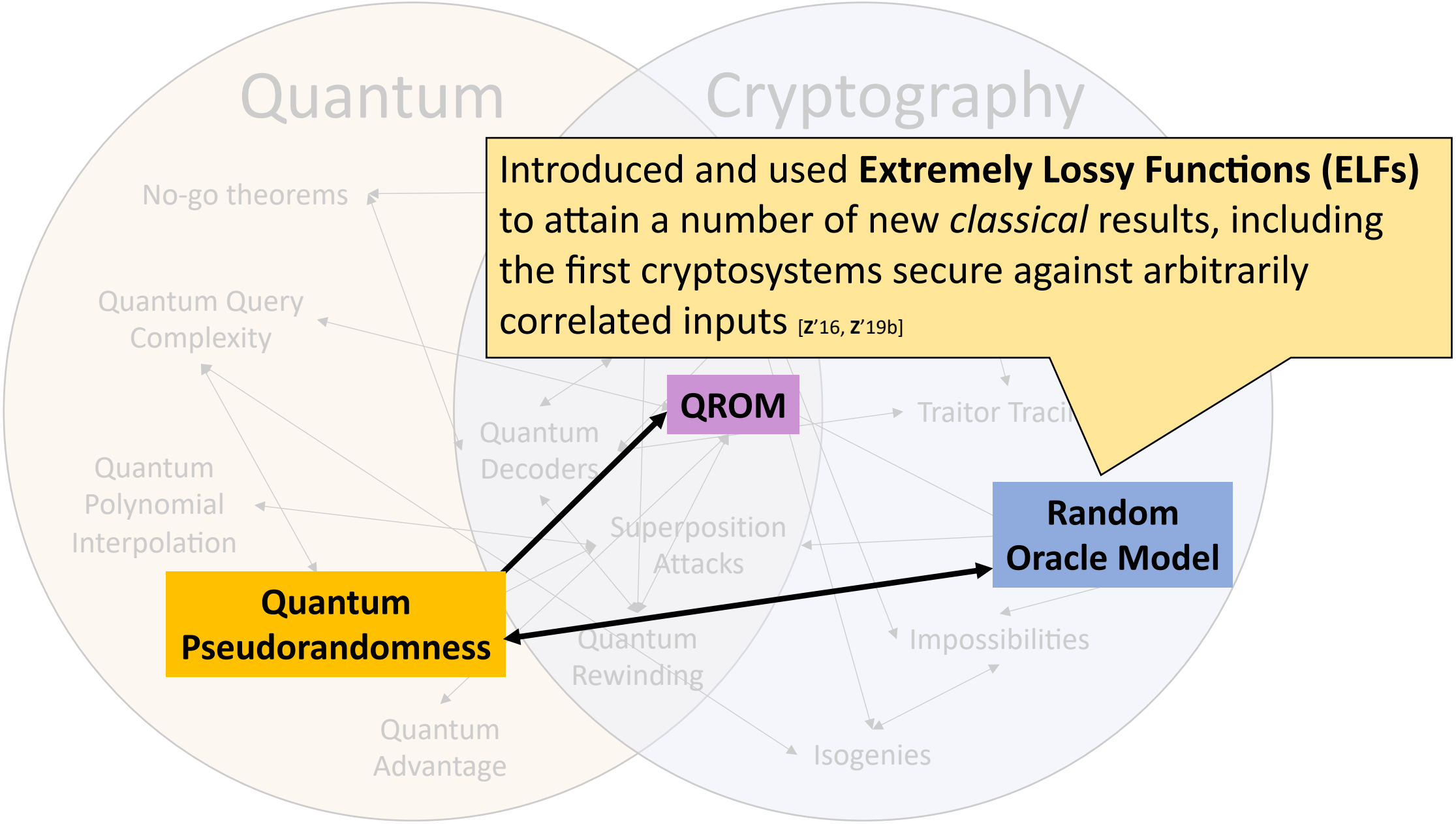
My work

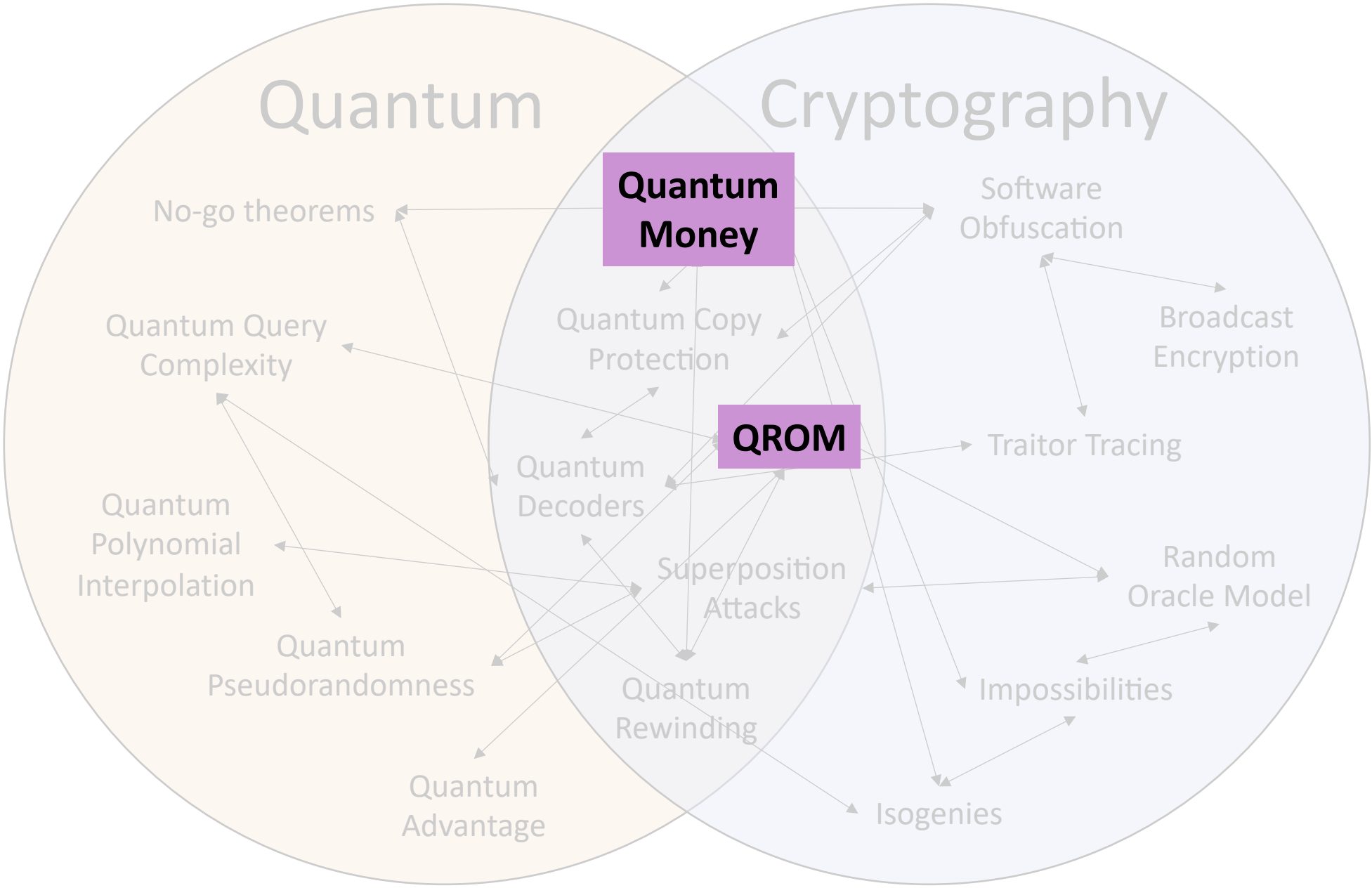

Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

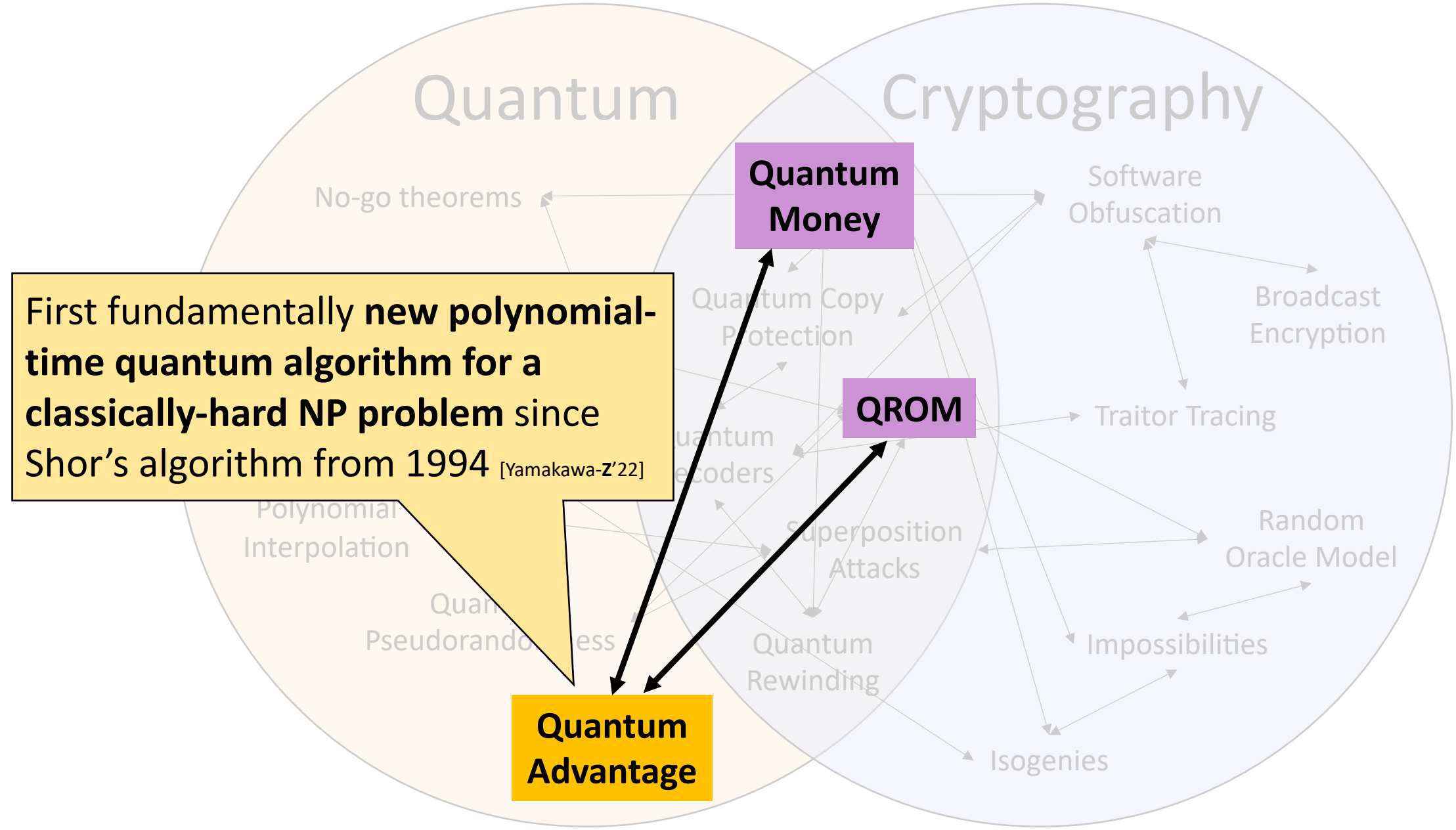


Initiated cryptographic study of the **Quantum Random Oracle Model (QROM)**, which has since become a major design consideration in the search for post-quantum cryptographic protocols [Boneh-Dagdelen-Fischlin-Lehmann-Schaffner-**Z**'11]

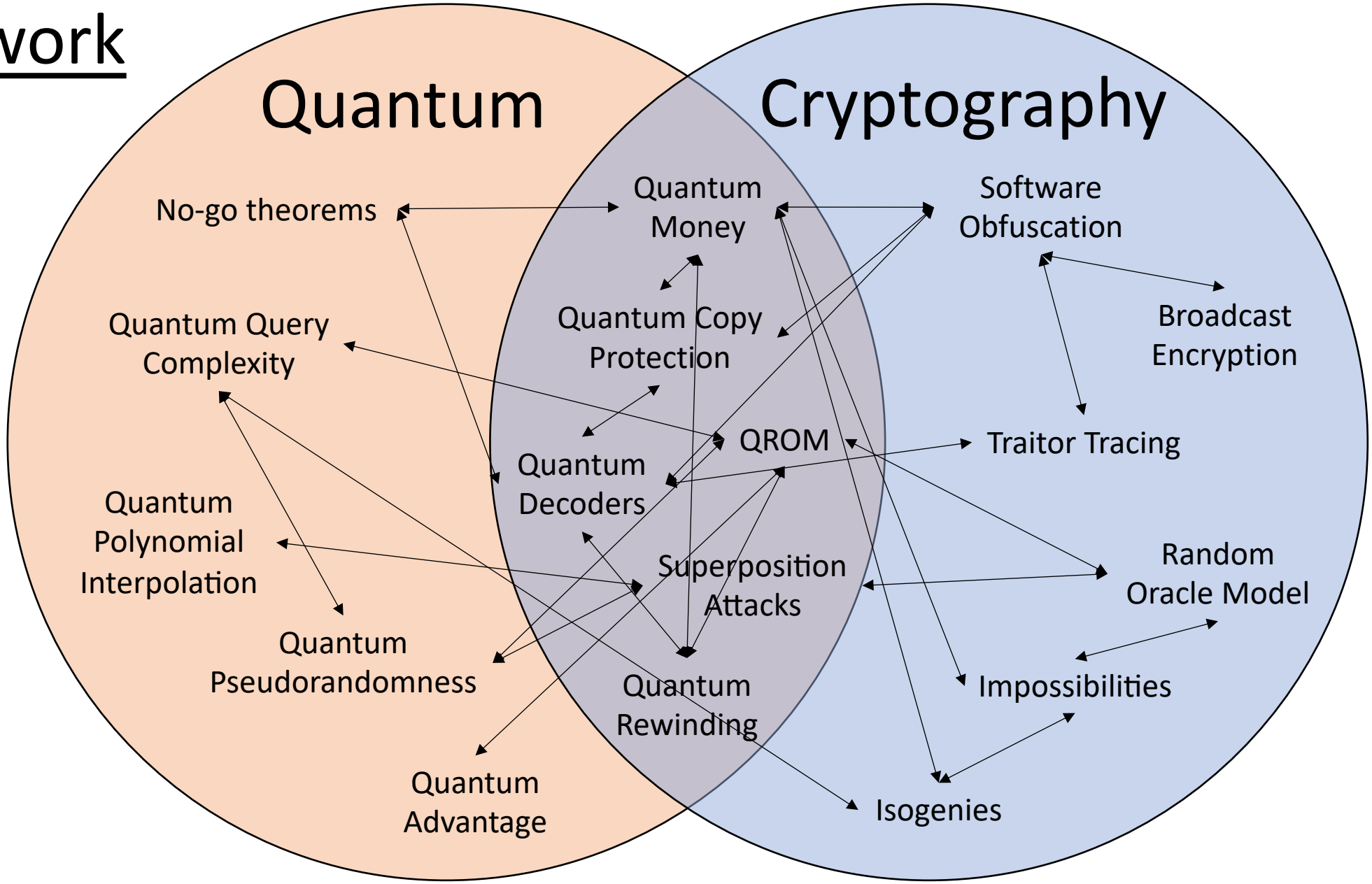

Quantum
Cryptography
First provably **quantum-secure pseudorandom functions (PRFs)** → central tool in quantum cryptography and complexity [Z'12]
Quantum Money
Software Obfuscation
Broadcast Encryption
Traitor Tracing
Random Oracle Model
Impossibilities
Isogenies
Quantum Decoders
Superposition Attacks
Quantum Rewinding
Quantum Advantage
Interpolation
QROM
Quantum Pseudorandomness

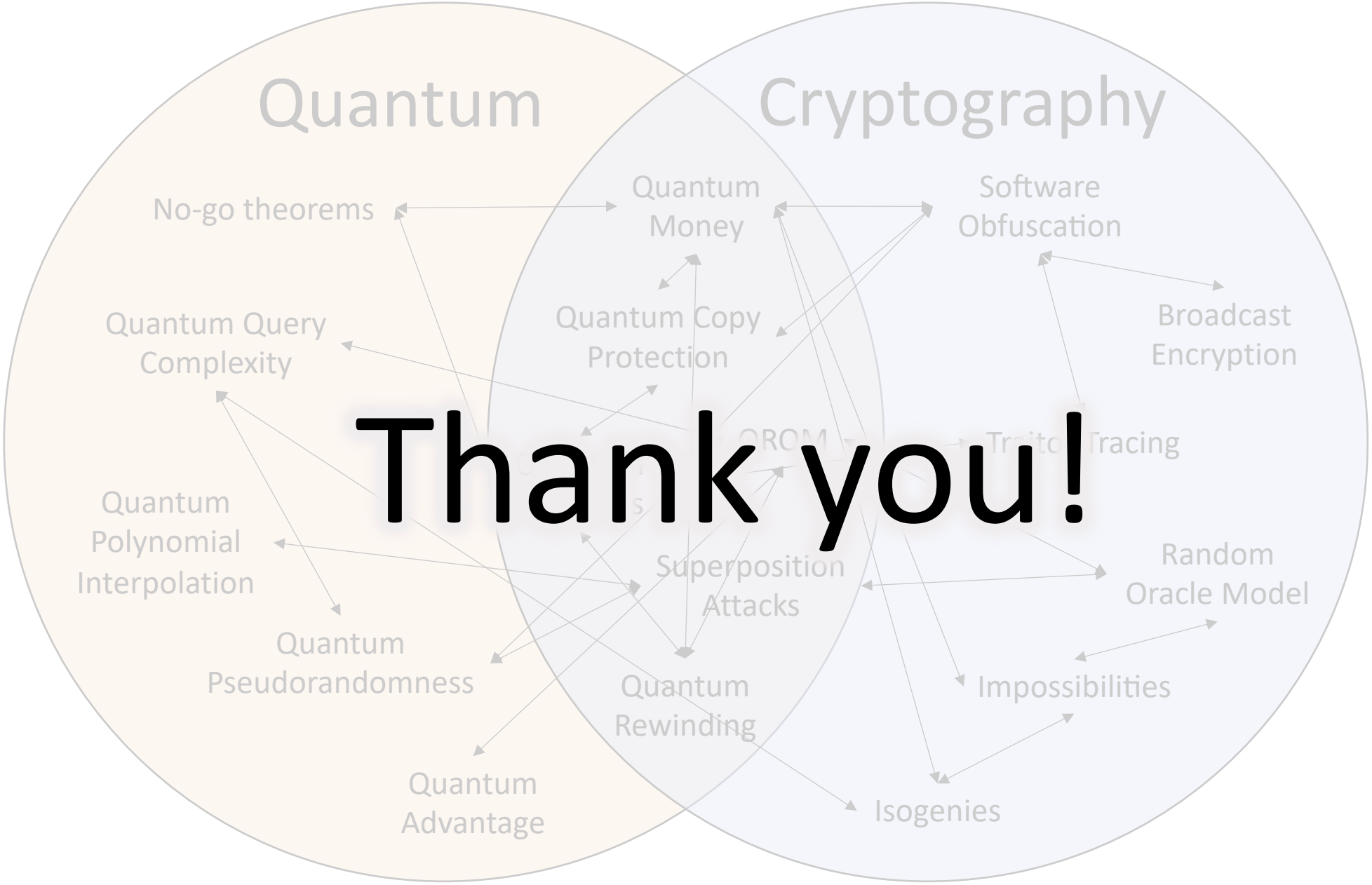

Quantum
Cryptography
No-go theorems
Quantum Query Complexity
Quantum Polynomial Interpolation
Quantum Decoders
Superposition Attacks
Quantum Rewinding
Quantum Advantage
Traitor Traci
Impossibilities
Isogenies
QROM
Quantum Pseudorandomness
Random Oracle Model
Introduced and used **Extremely Lossy Functions (ELFs)** to attain a number of new *classical* results, including the first cryptosystems secure against arbitrarily correlated inputs [Z'16, Z'19b]


Quantum
Cryptography
Quantum Money
QROM
No-go theorems
Quantum Query Complexity
Quantum Polynomial Interpolation
Quantum Pseudorandomness
Quantum Advantage
Quantum Copy Protection
Quantum Decoders
Superposition Attacks
Quantum Rewinding
Software Obfuscation
Broadcast Encryption
Traitor Tracing
Random Oracle Model
Impossibilities
Isogenies


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Rewinding
Impossibilities
Isogenies
Quantum Advantage
First fundamentally **new polynomial-time quantum algorithm for a classically-hard NP problem** since Shor's algorithm from 1994 [Yamakawa-Z'22]

My work


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
QROM
Traitor Tracing
Quantum Decoders
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies


Quantum
Cryptography
No-go theorems
Quantum Money
Software Obfuscation
Quantum Query Complexity
Quantum Copy Protection
Broadcast Encryption
Quantum Polynomial Interpolation
Superposition Attacks
Random Oracle Model
Quantum Pseudorandomness
Quantum Rewinding
Impossibilities
Quantum Advantage
Isogenies

Thank you!