SQUIRRELS

Square
Unstructured
IntegeRR
Euclidean
Lattice
Signature

A PROPOSAL FOR NIST IV CALL ON SIGNATURES

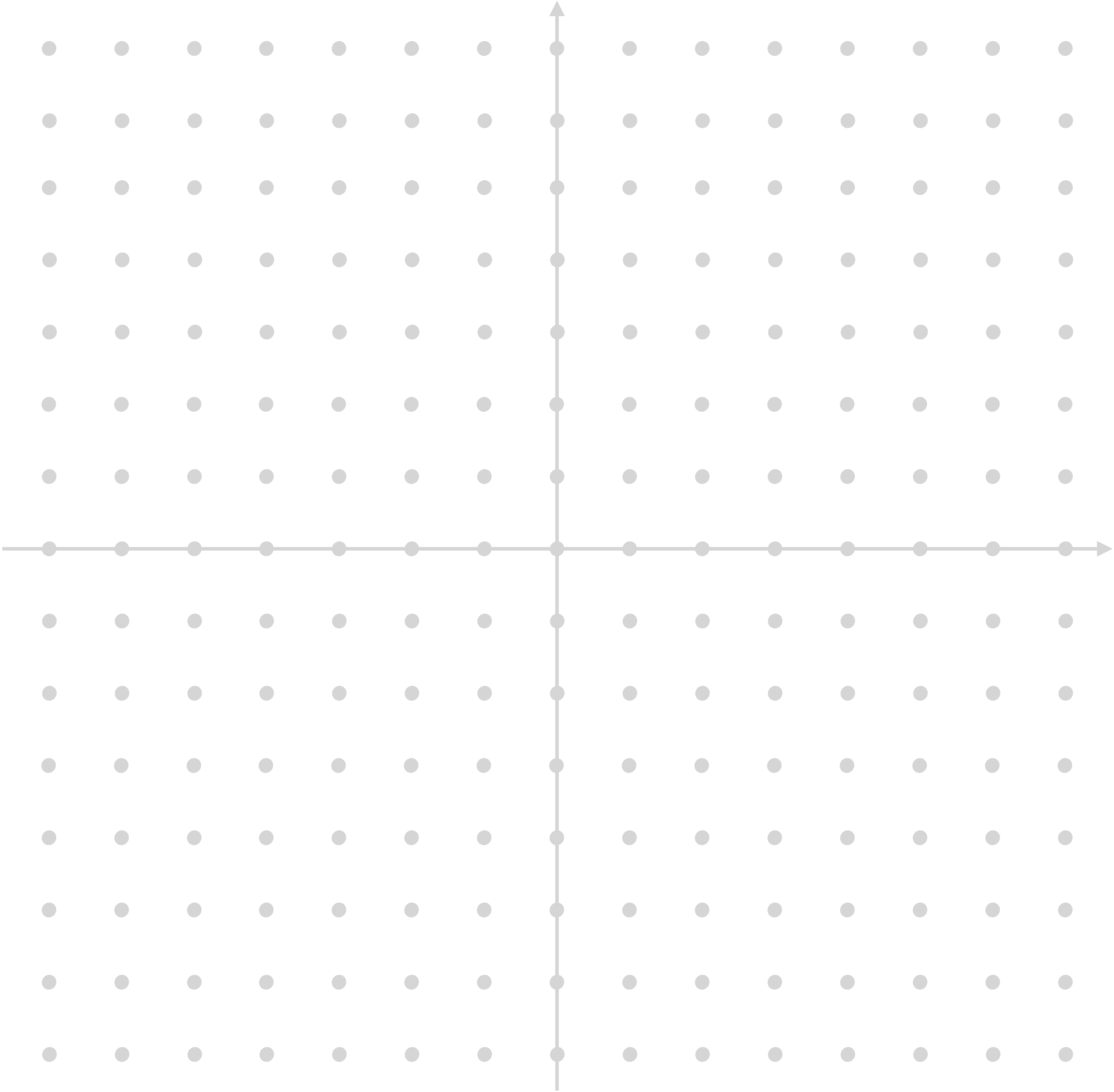
T.ESPITAU, GUILHEM NIOT, SUN CHAO, M. TIBOUCHI
PQSHIELD PQSHIELD, ENS LYON OSAKA UNIVERSITY NTT LABORATORIES

A panorama of signatures (sizes)

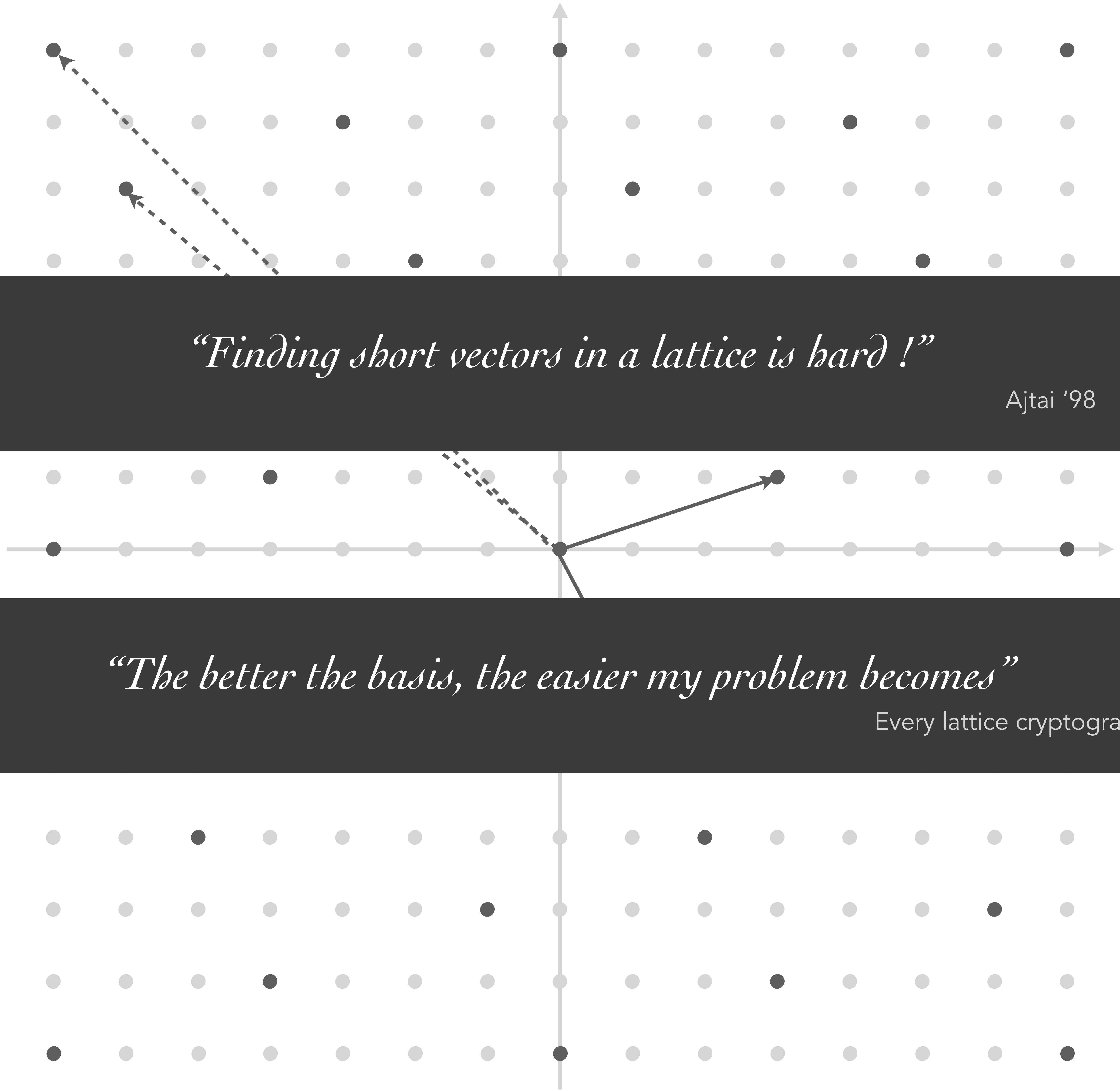
PICNIC
SPHINCS+
R S A
2048
4096
DILITHIUM
XMSS
ED25519
FALCON

SMALL … BUT ALSO UNSTRUCTURED

Lattice



Lattice



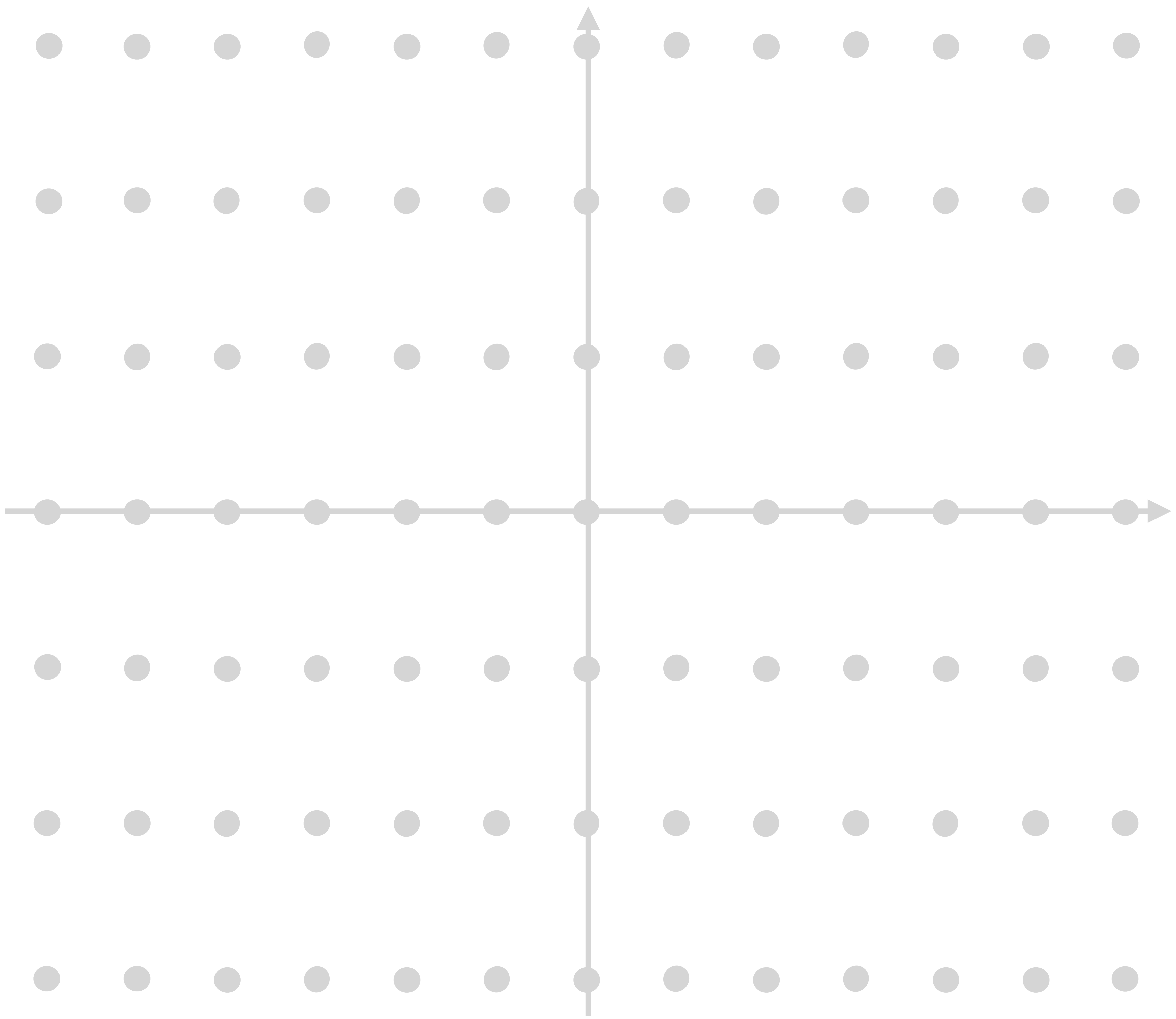
> *"Finding short vectors in a lattice is hard !"*
>
> Ajtai '98

> *"The better the basis, the easier my problem becomes"*
>
> Every lattice cryptographer ever

Hash-and-sign over lattices 101

```
Sign (sk, msg)
  1.$\mathbf{m}$ <— Hash (msg)
  2.$\mathbf{v}$ <— Discrete Gaussian sample ($\mathbf{m}$)
  3.Return $\mathbf{s}$ = ($\mathbf{m}$-$\mathbf{v}$)
```

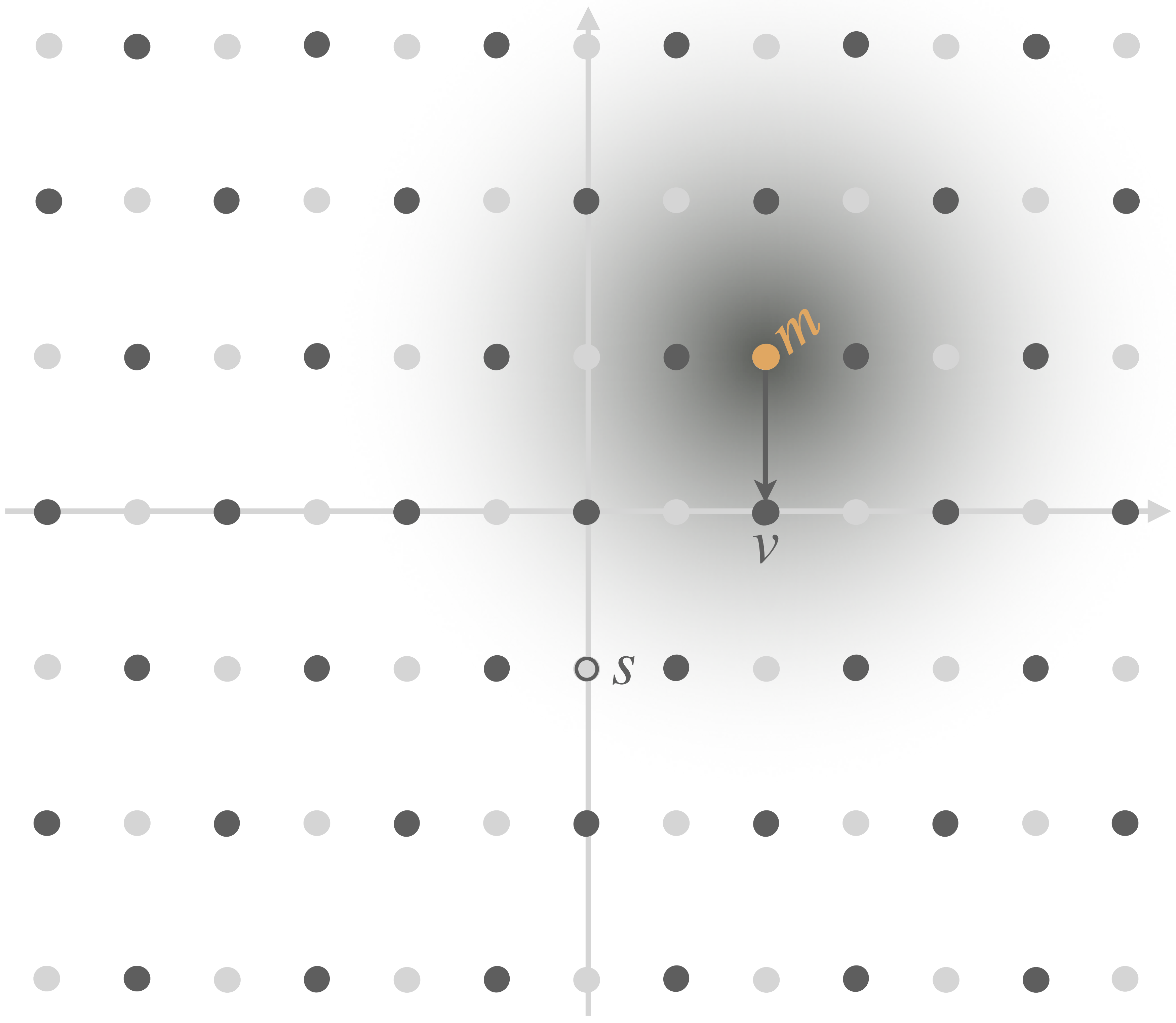
```
Verif (pk, msg, s)
  1.Assert $||\mathbf{s}||$ small
  2.Assert $\mathbf{s}$-Hash(msg) is in L
  3.Accept
```



Hash-and-sign over lattices 101

```
Sign (sk, msg)
1.m <— Hash (msg)
2.v <— Discrete Gaussian sample (m)
3.Return s = (m-v)
```

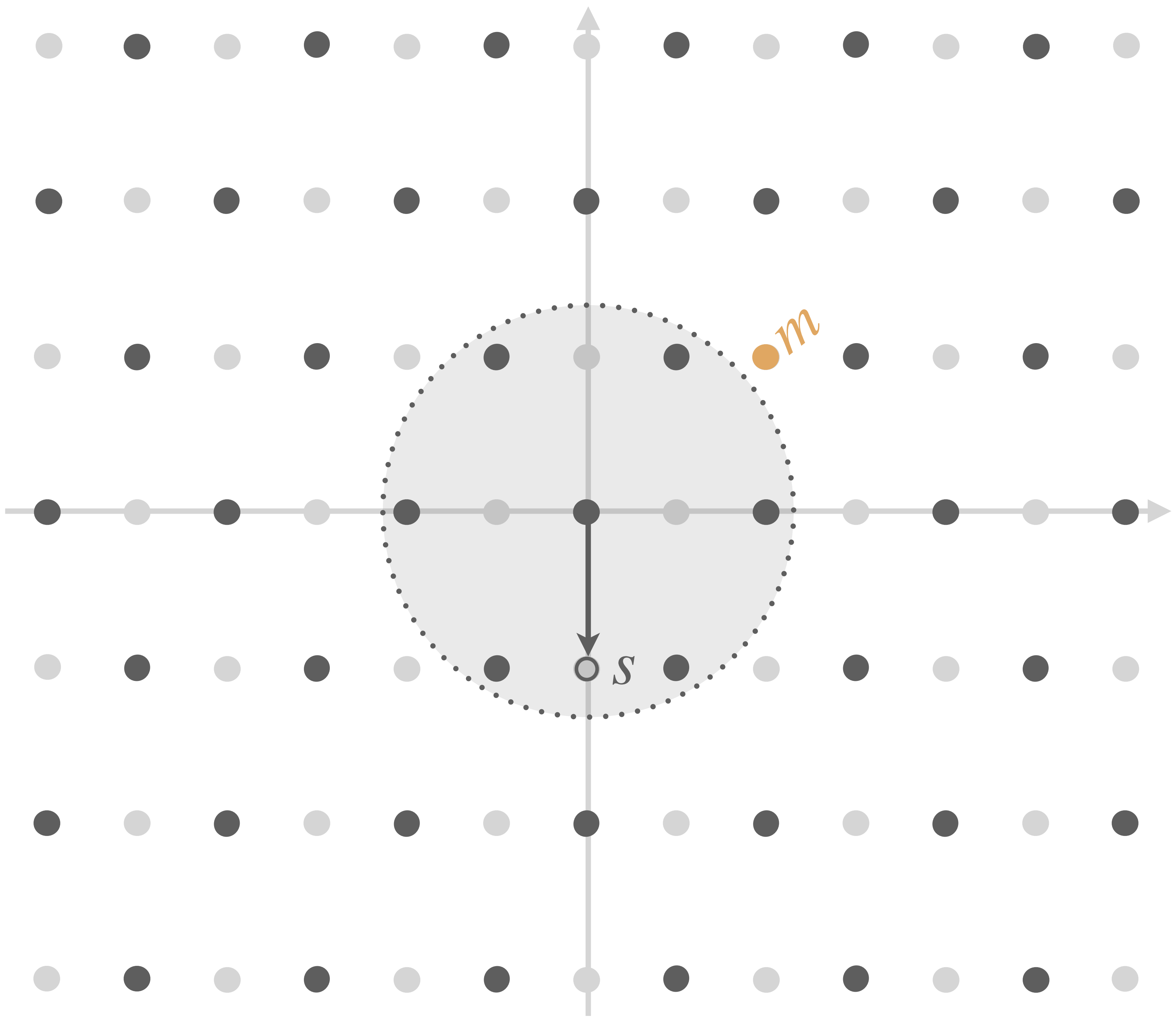
```
Verif (pk, msg, s)
1.Assert ||s|| small
2.Assert s-Hash(msg) is in L
3.Accept
```



Hash-and-sign over lattices 101

```
Sign (sk, msg)
1.m <– Hash (msg)
2.v <– Discrete Gaussian sample (m)
3.Return s = (m-v)
```

```
Verif (pk, msg, s)
1.Assert ||s|| small
2.Assert s-Hash(msg) is in L
3.Accept
```



Hash-and-sign over lattices 101

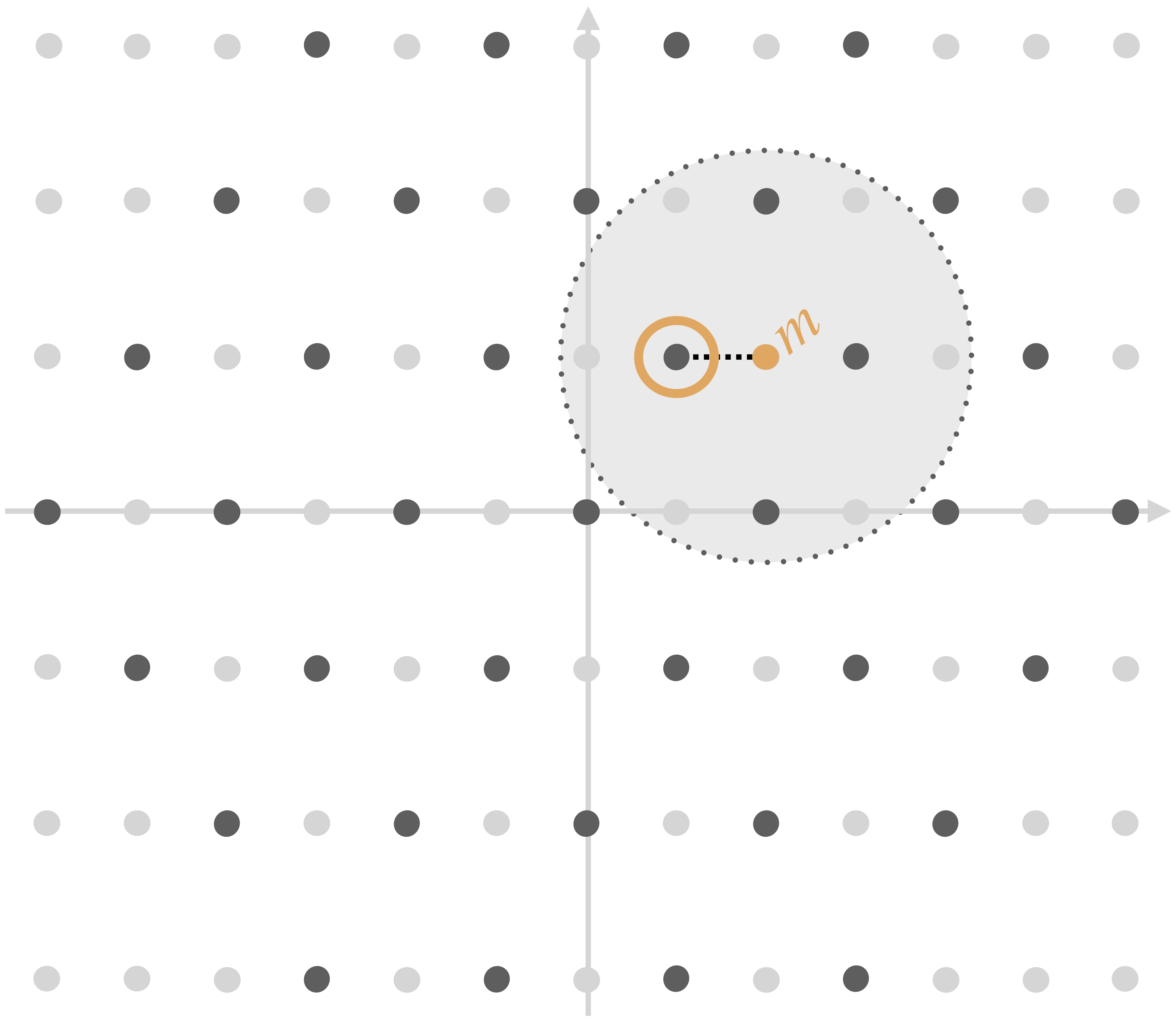
FORGING A SIGNATURE

Find a lattice point close to the hash

Verification: check that

1. candidate is inside L
2. close to hash

[Closest Vector Problem (CVP) instance]



"The better the basis, the easier my problem becomes"

Every lattice cryptographer ever

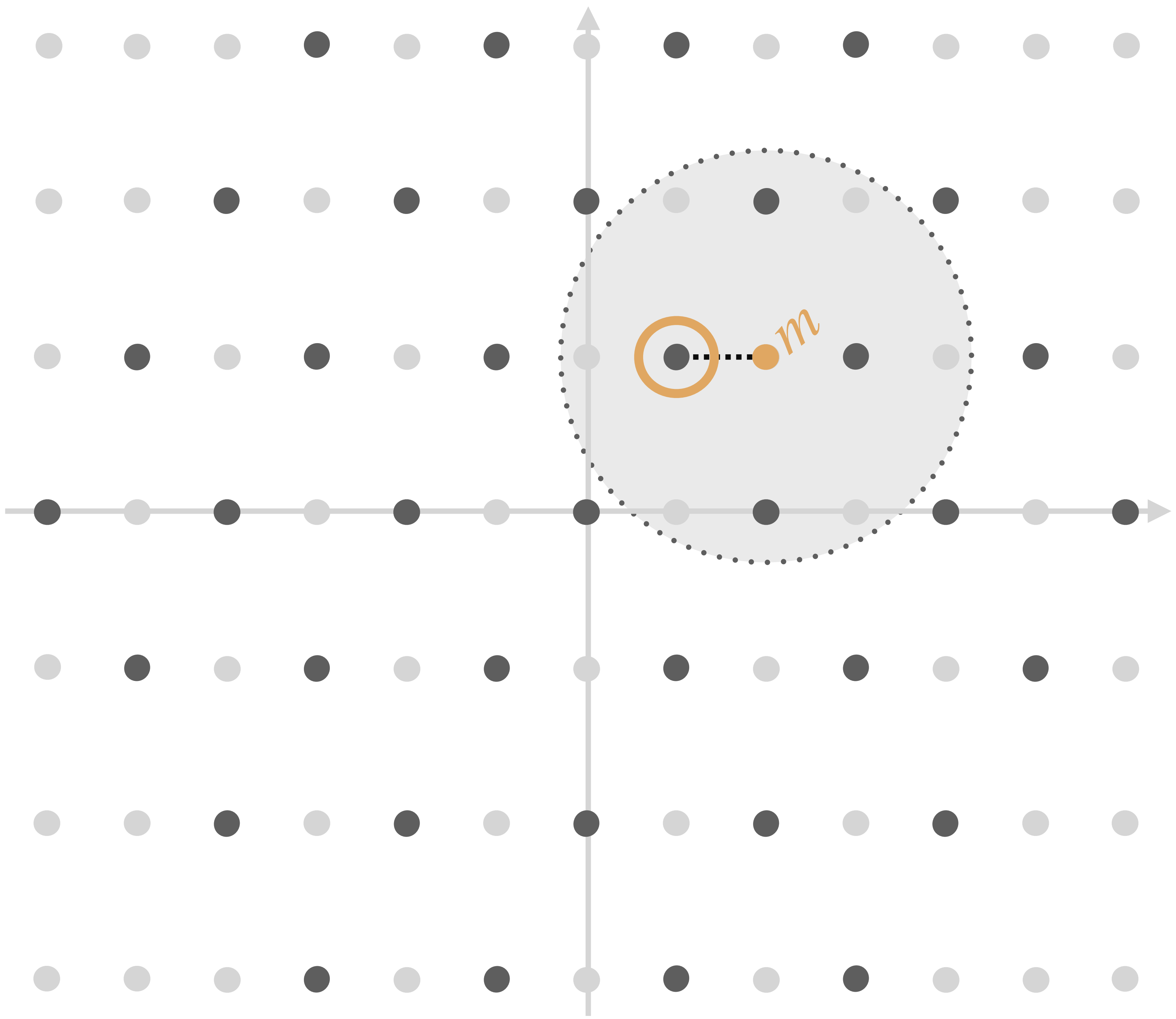
Hash-and-sign over lattices 101

FORGING A SIGNATURE

Find a lattice point close to the hash

Should be *hard*:

- > small distance gaussian (= small variance)
- > 'good private basis (= short vectors)



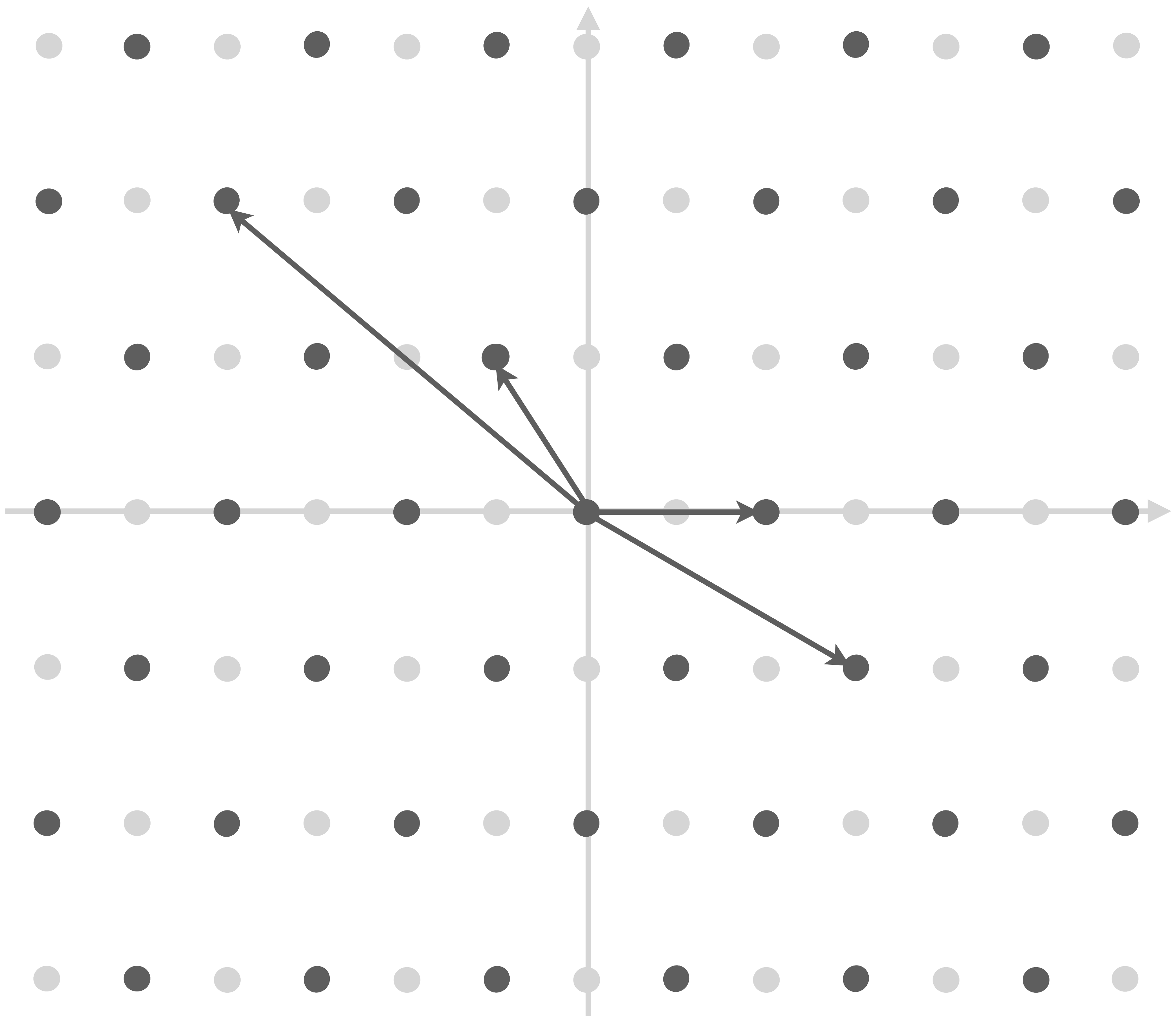
> *"The better the basis, the easier my problem becomes"*
>
> Every lattice cryptographer ever

Hash-and-sign over lattices 101

KEY RECOVERY *Find the secret key directly*

Lattice reduction / SVP (find short vectors)

Goes from public lattice to short vectors



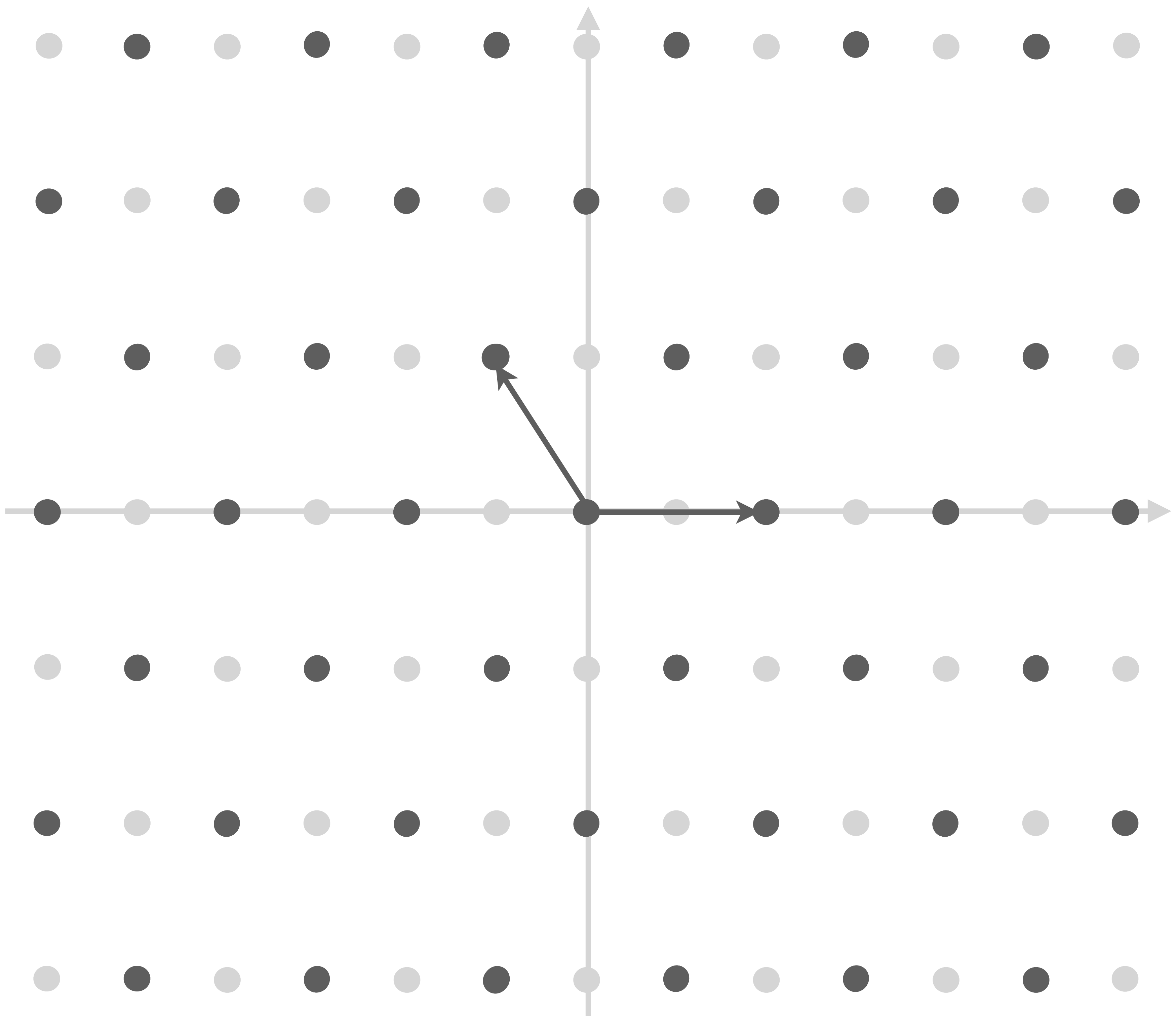
“Finding short vectors in a lattice is hard !”
Ajtai ‘98

Hash-and-sign over lattices 101

KEY RECOVERY *Find the secret key directly*

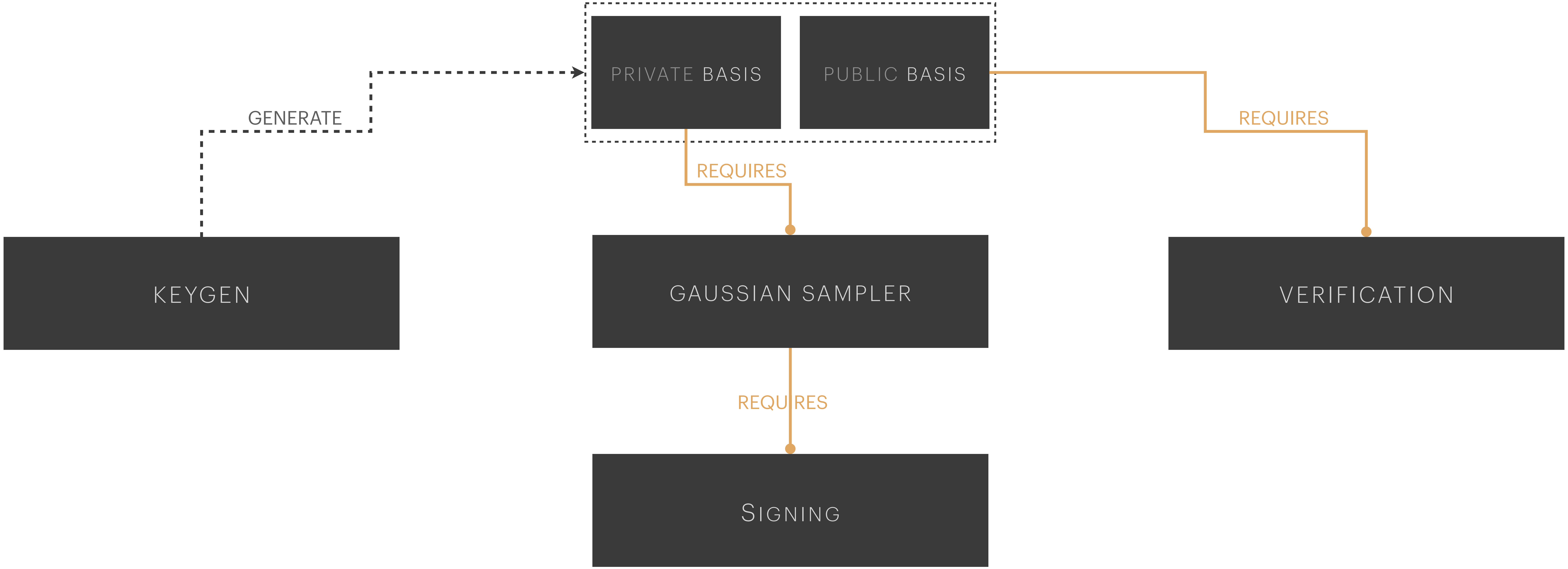
Should be hard:

- > large dimension (> hard reduction)
- > bad private basis (= long vectors)

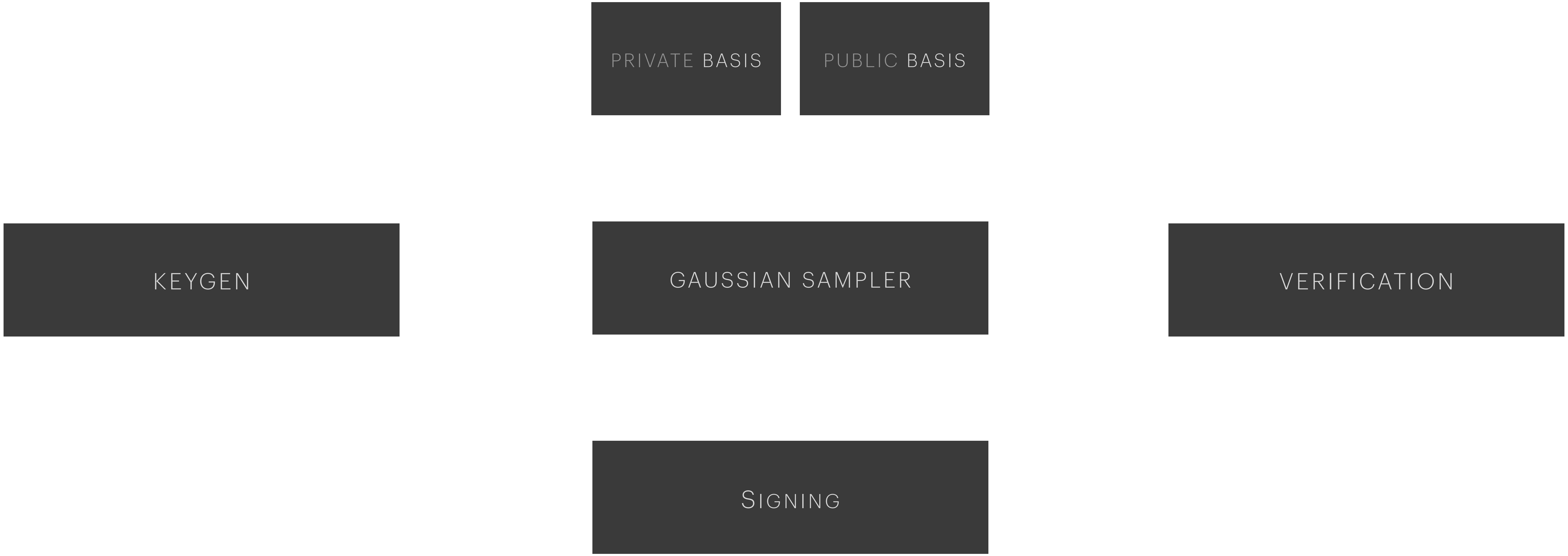


"Finding short vectors in a lattice is hard !"
Ajtai '98

Design rationale



Design rationale



PRIVATE BASIS

PUBLIC BASIS

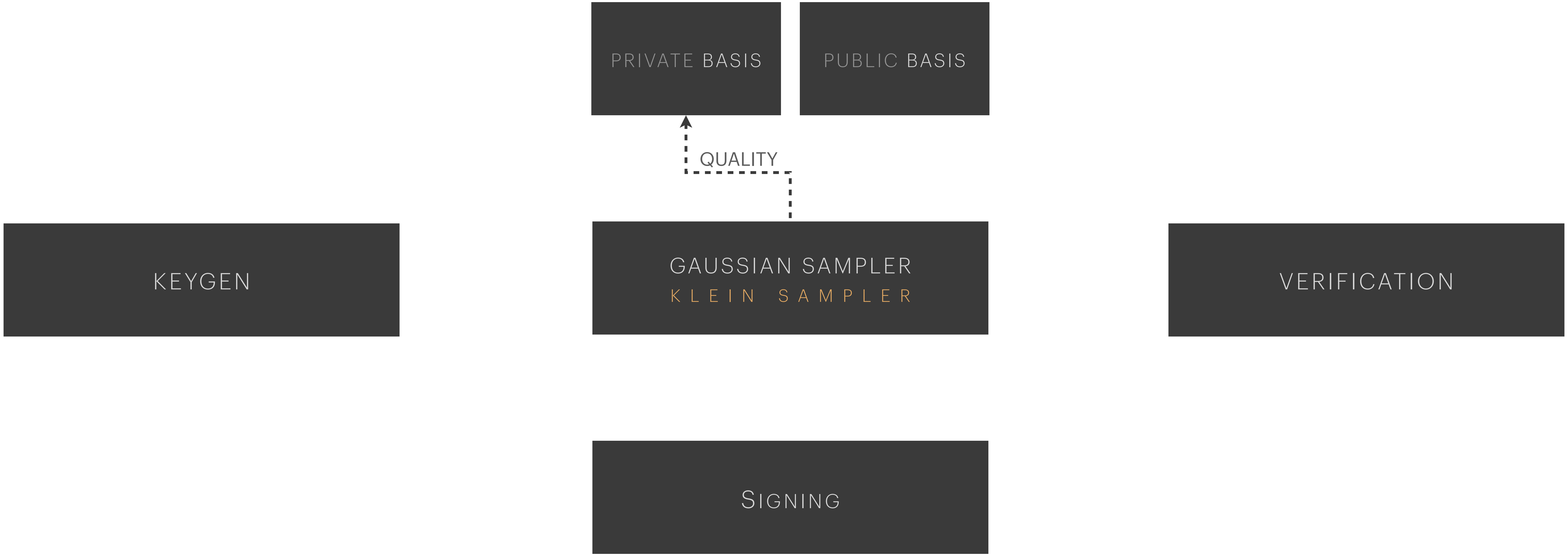
KEYGEN

GAUSSIAN SAMPLER

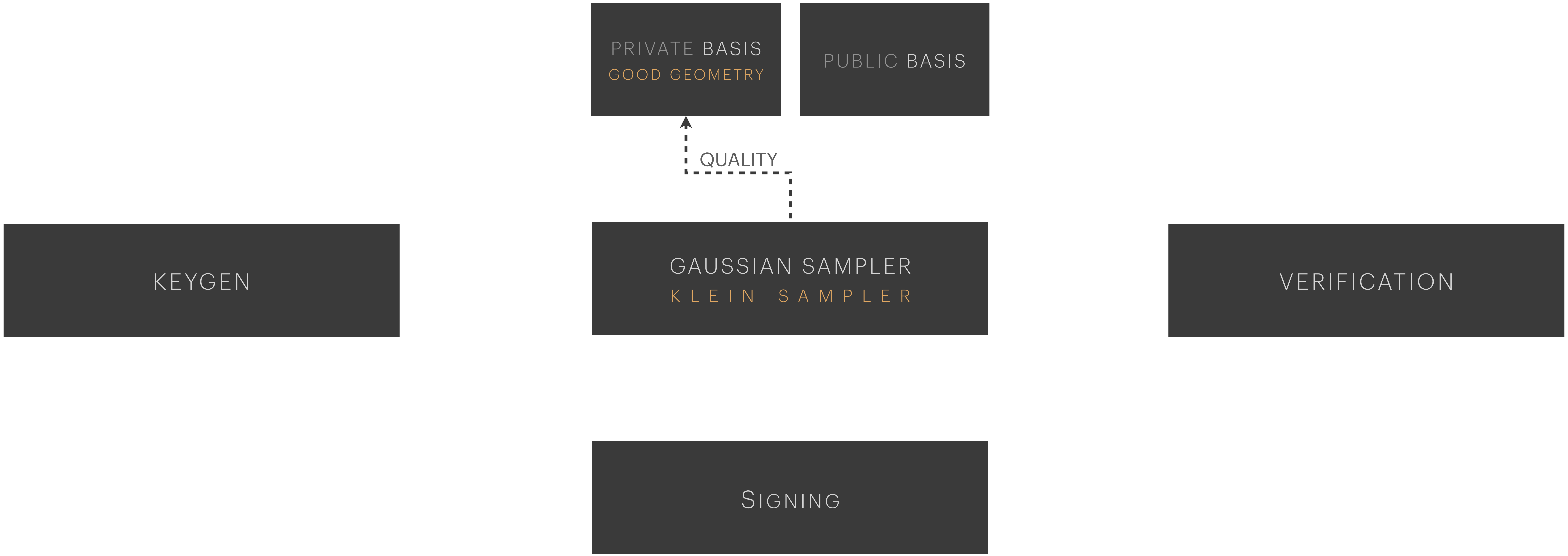
VERIFICATION

SIGNING

Design rationale



Design rationale



PRIVATE BASIS

GOOD GEOMETRY

PUBLIC BASIS

QUALITY

KEYGEN

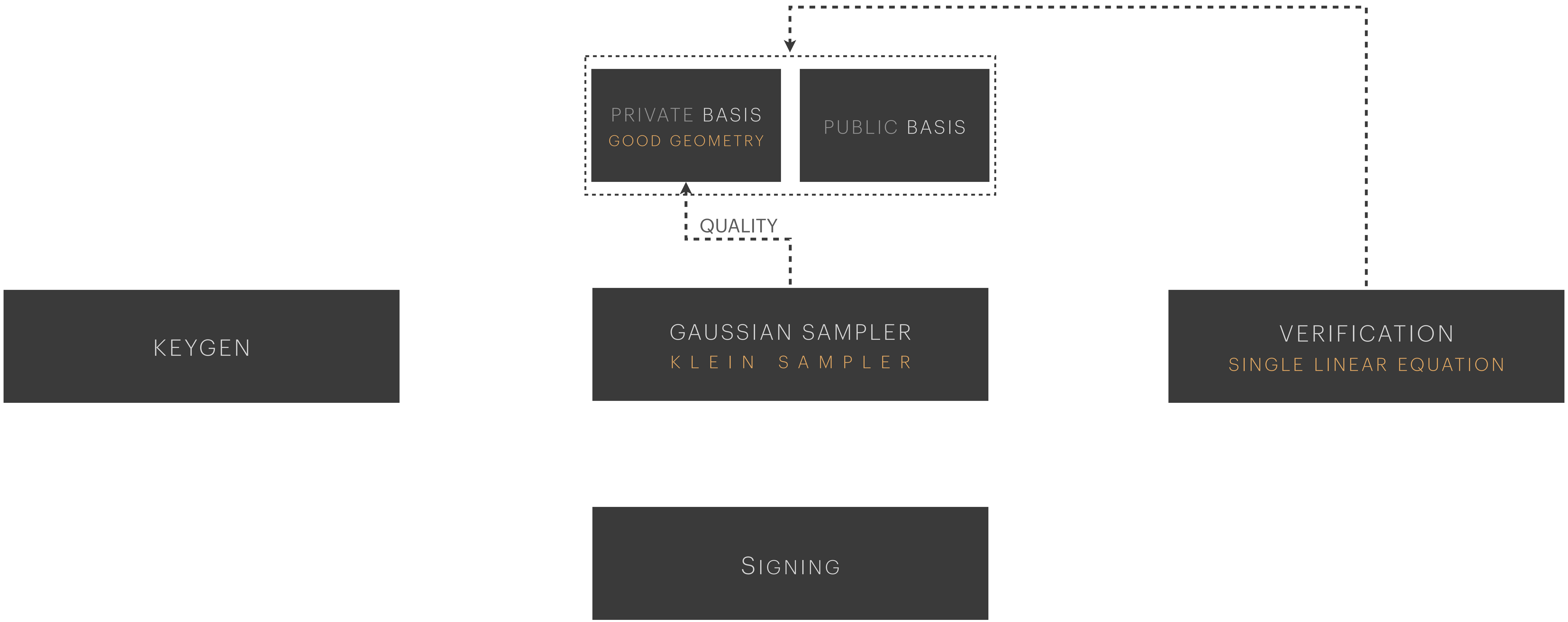
GAUSSIAN SAMPLER

KLEIN SAMPLER

VERIFICATION

SIGNING

Design rationale



PRIVATE BASIS

GOOD GEOMETRY

PUBLIC BASIS

QUALITY

KEYGEN

GAUSSIAN SAMPLER

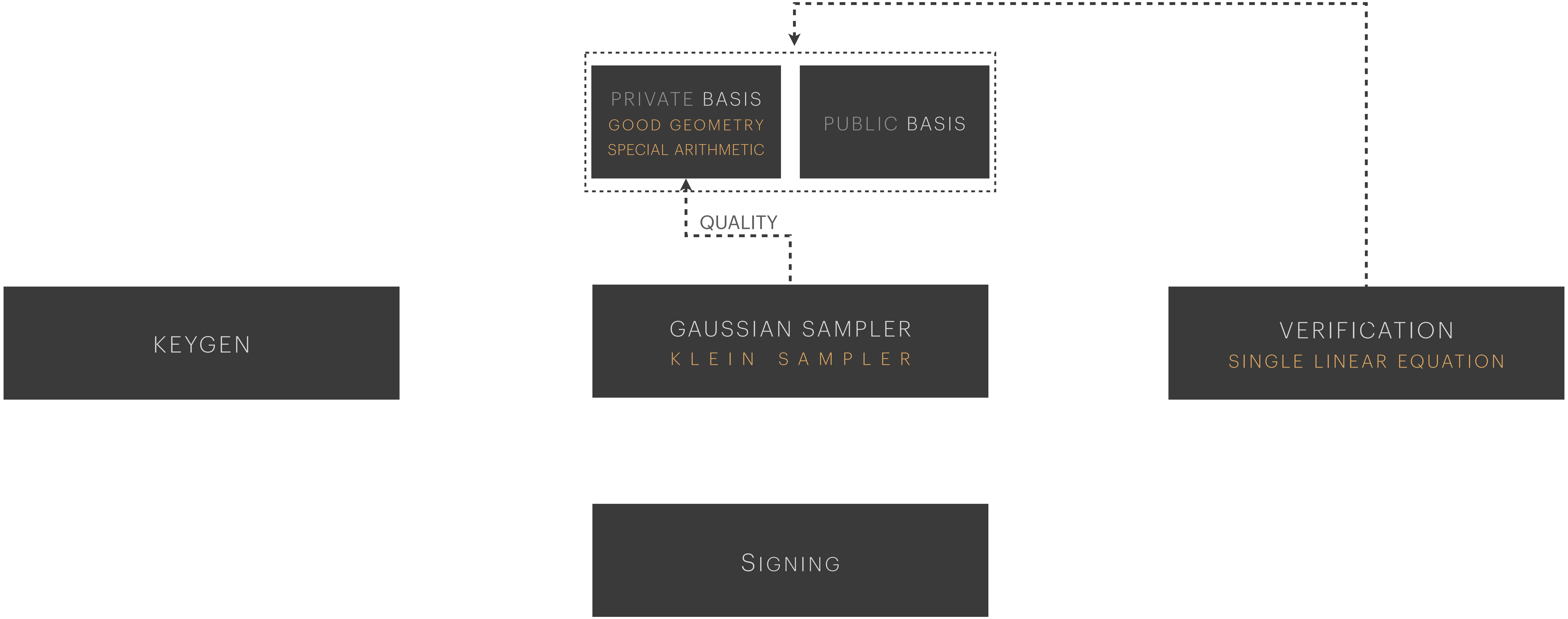
KLEIN SAMPLER

VERIFICATION

SINGLE LINEAR EQUATION

SIGNING

Design rationale



PRIVATE BASIS

GOOD GEOMETRY

SPECIAL ARITHMETIC

PUBLIC BASIS

QUALITY

KEYGEN

GAUSSIAN SAMPLER

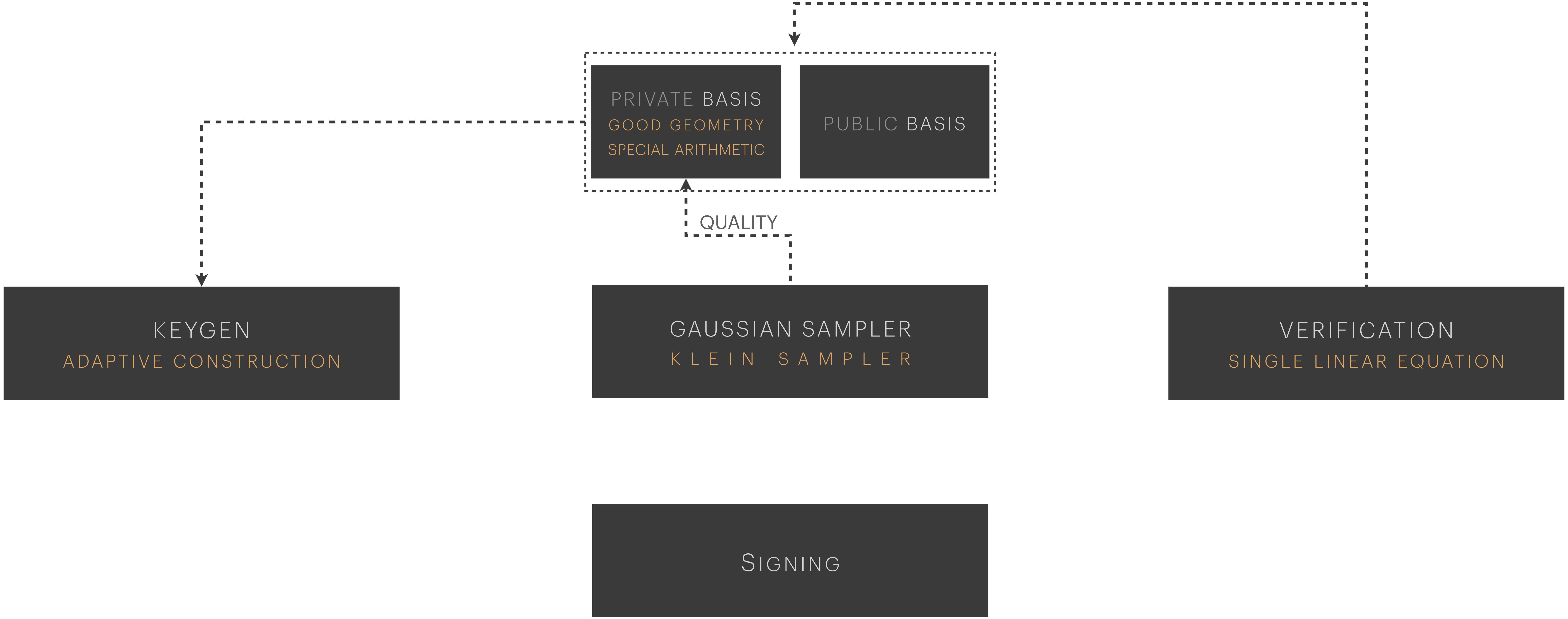
KLEIN SAMPLER

VERIFICATION

SINGLE LINEAR EQUATION

SIGNING

Design rationale



PRIVATE BASIS

GOOD GEOMETRY

SPECIAL ARITHMETIC

PUBLIC BASIS

QUALITY

KEYGEN

ADAPTIVE CONSTRUCTION

GAUSSIAN SAMPLER

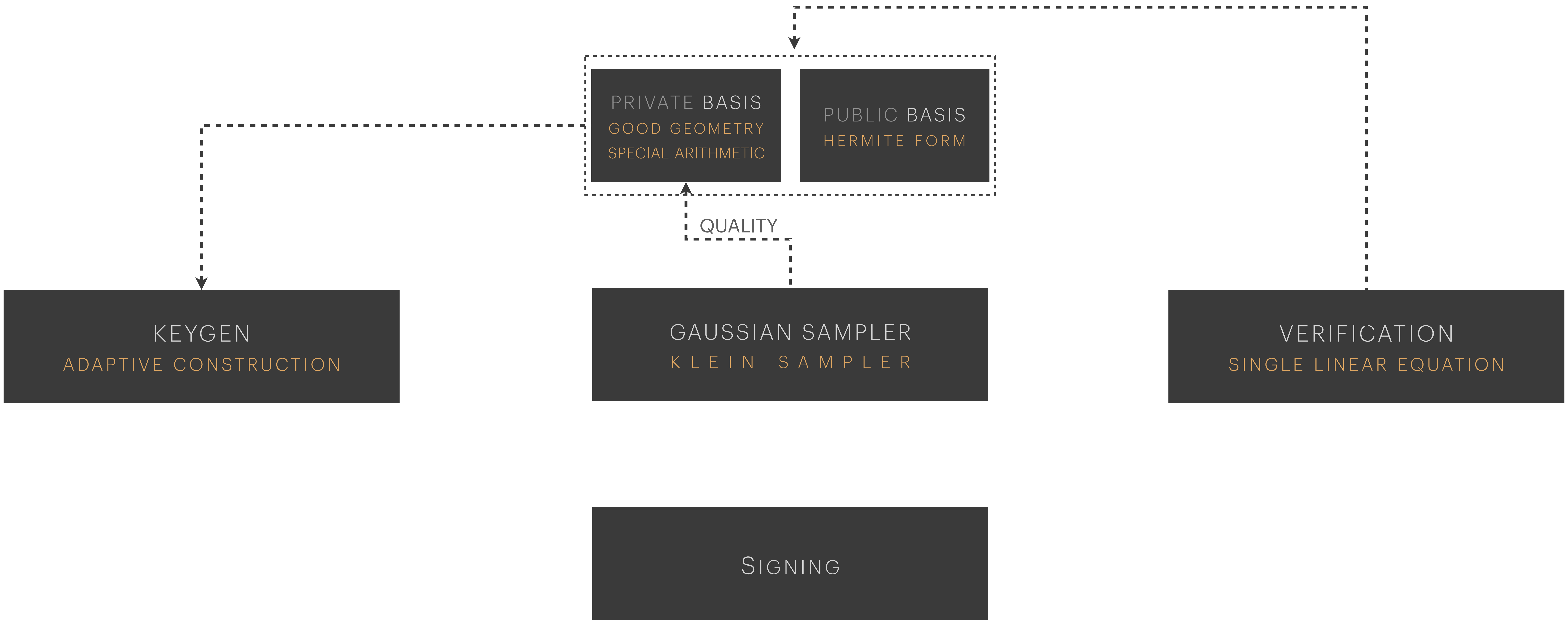
KLEIN SAMPLER

VERIFICATION

SINGLE LINEAR EQUATION

SIGNING

Design rationale



PRIVATE BASIS

GOOD GEOMETRY

SPECIAL ARITHMETIC

PUBLIC BASIS

HERMITE FORM

QUALITY

KEYGEN

ADAPTIVE CONSTRUCTION

GAUSSIAN SAMPLER

KLEIN SAMPLER

VERIFICATION

SINGLE LINEAR EQUATION

SIGNING

What does “good” means?

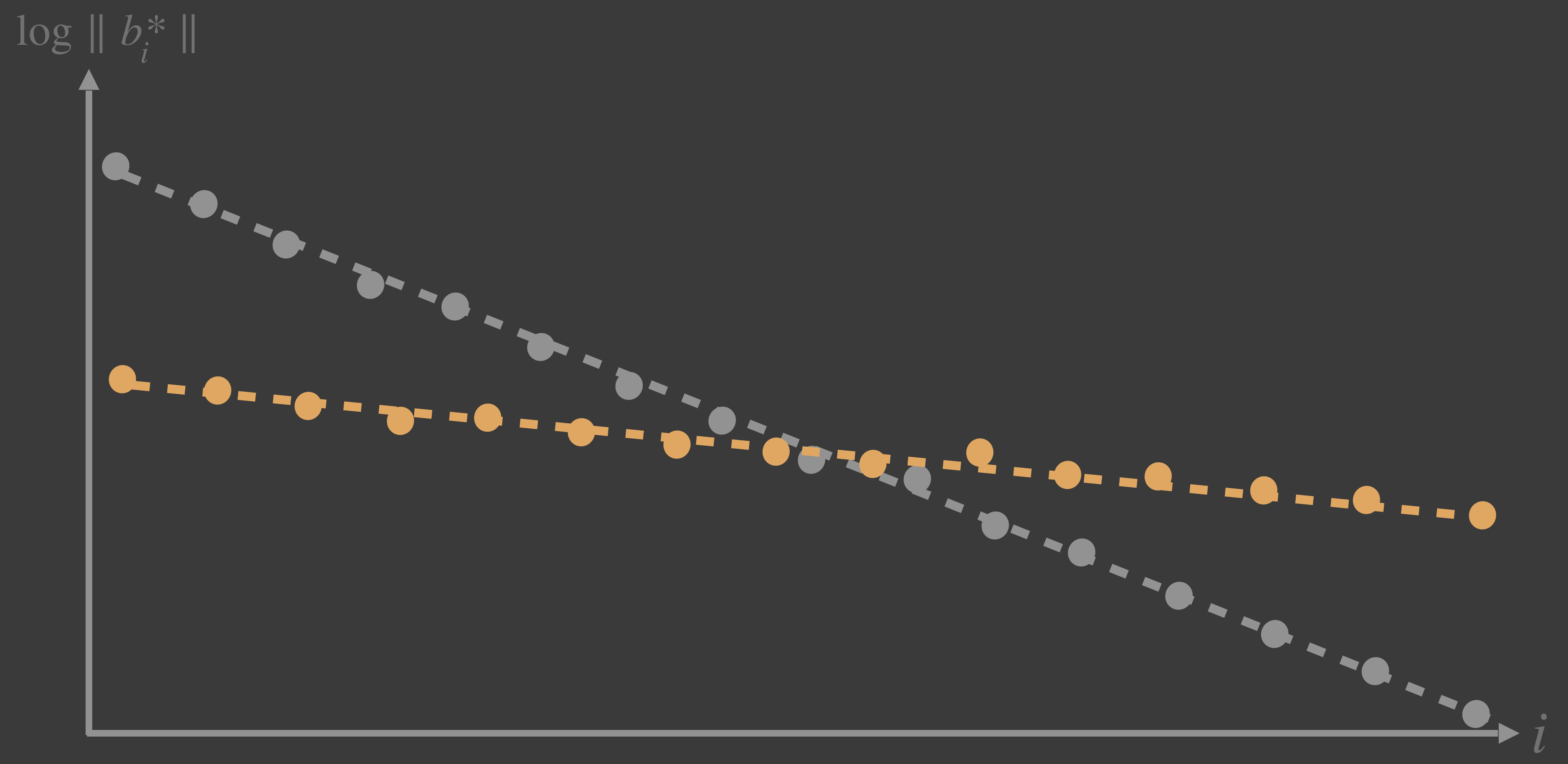
GOOD GEOMETRIC STRUCTURE

[flat basis profile]

Klein sampler’s quality $\propto$ max Gram-Schmidt norms

> low decay

> construct one vector after another by sampling in the good corresponding region of the space



GOOD ARITHMETIC STRUCTURE

[cyclic quotient structure]

“Co-representation” of integer lattice as **ker** of a map

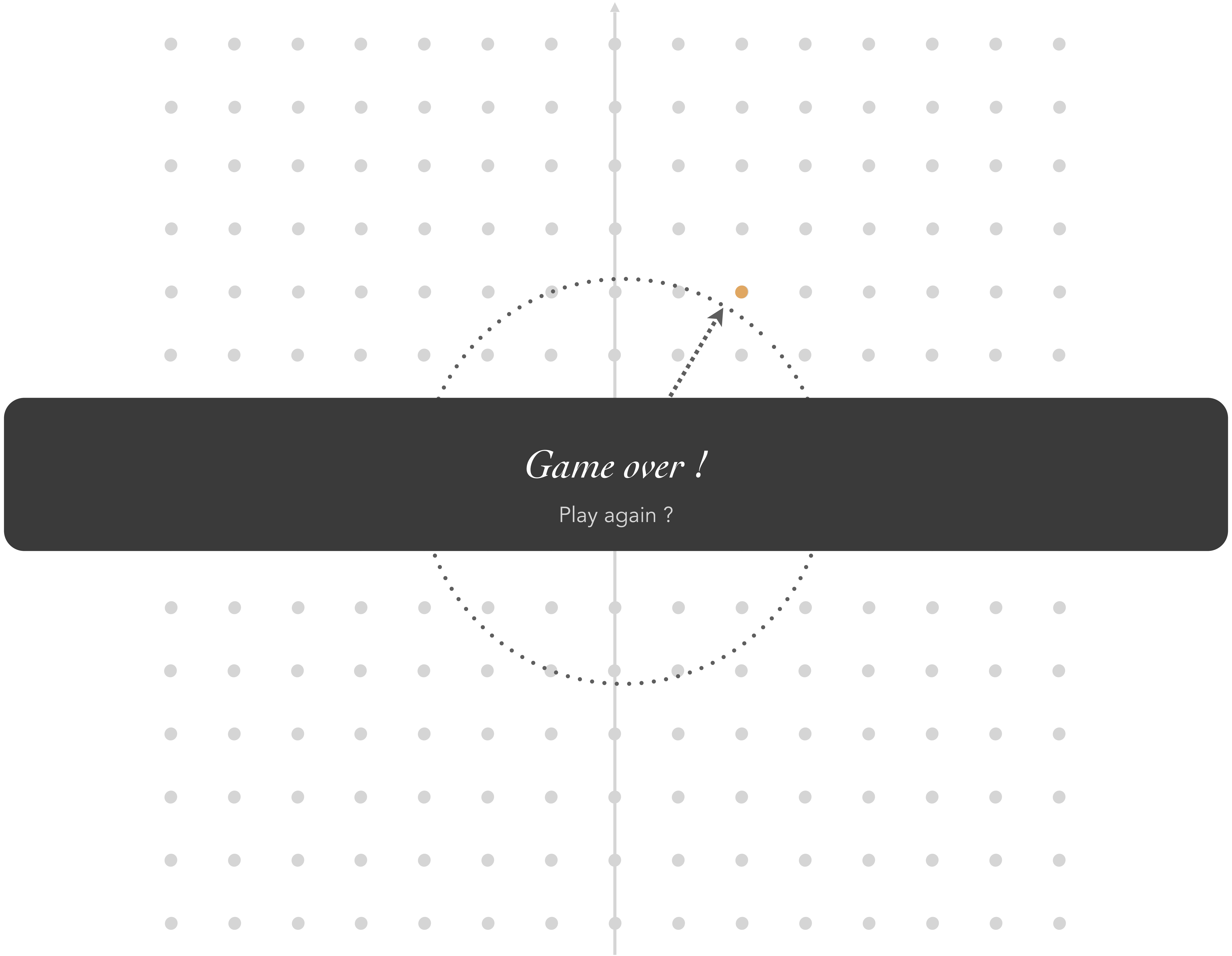
$$A : \mathbb{Z}^n \to (\mathbb{Z}/q\mathbb{Z})^m$$

$$v \in \mathscr{L} \Leftrightarrow Av = 0 \pmod q$$

> $m = 1$: *single equation mod q* ! $\langle v, \underline{a} \rangle = 0 \pmod q$

Cocyclic lattices — enforced by forcing the det to be squarefree

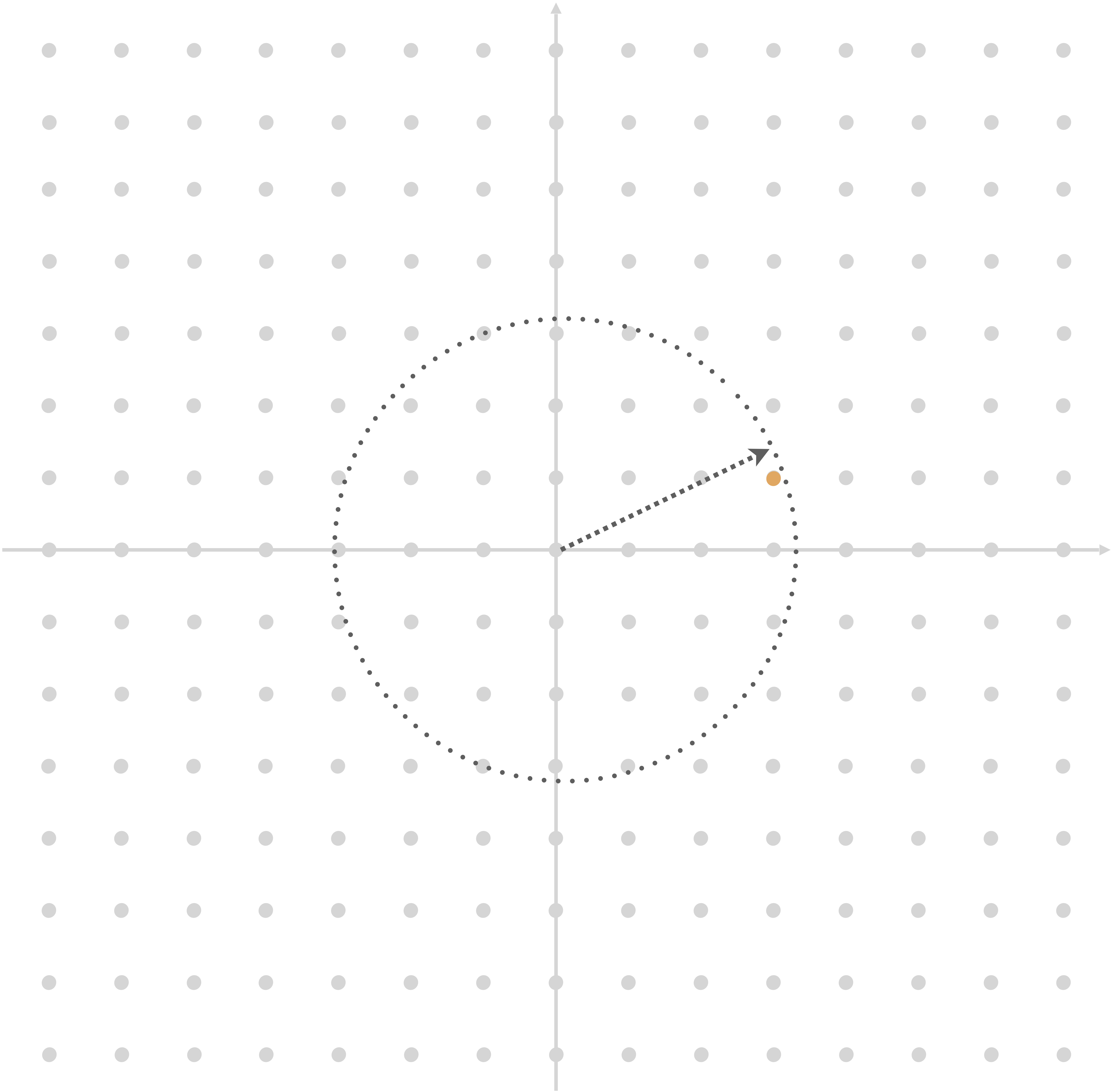
Dimension 2 example



Game over !

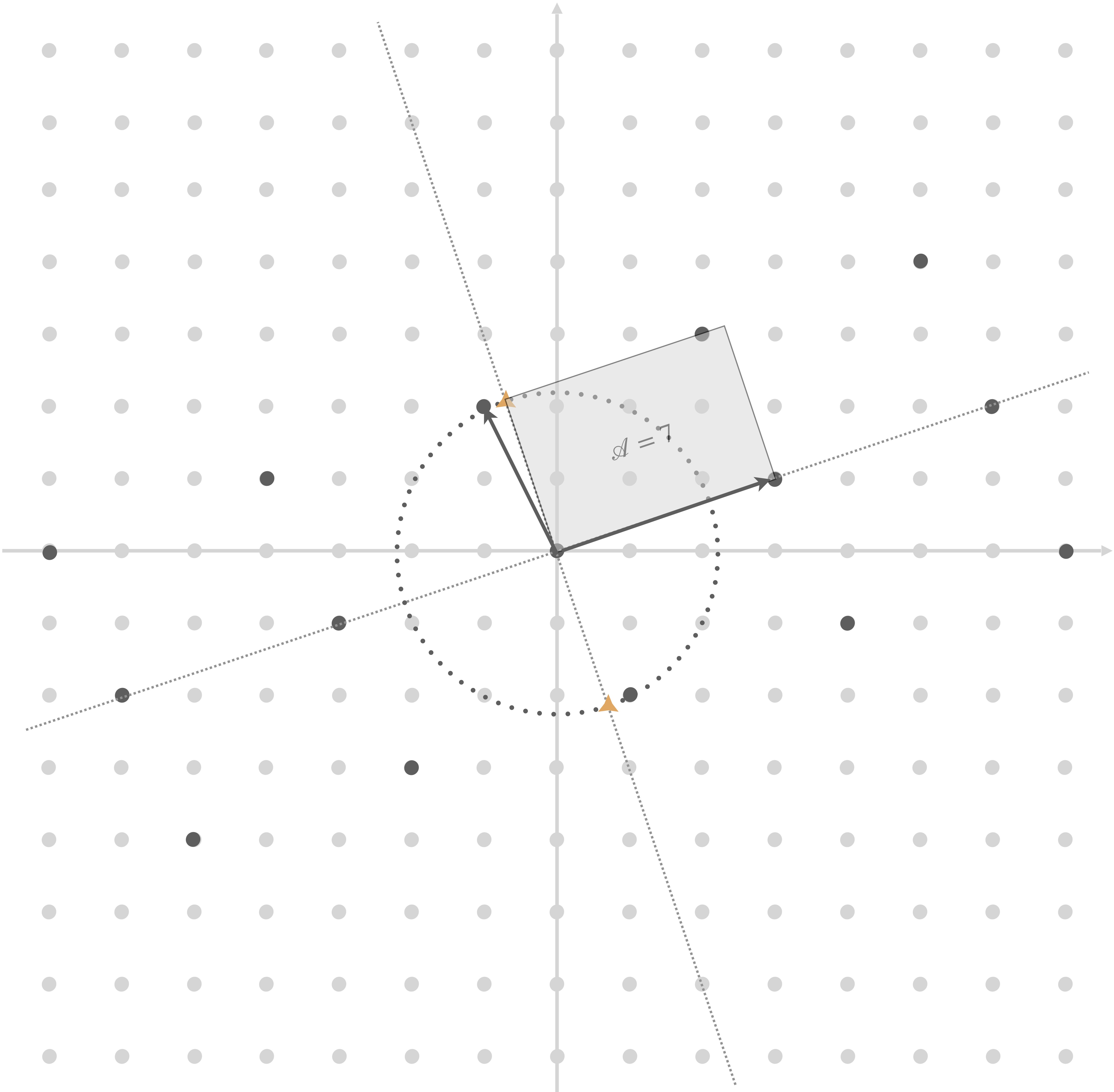
Play again ?

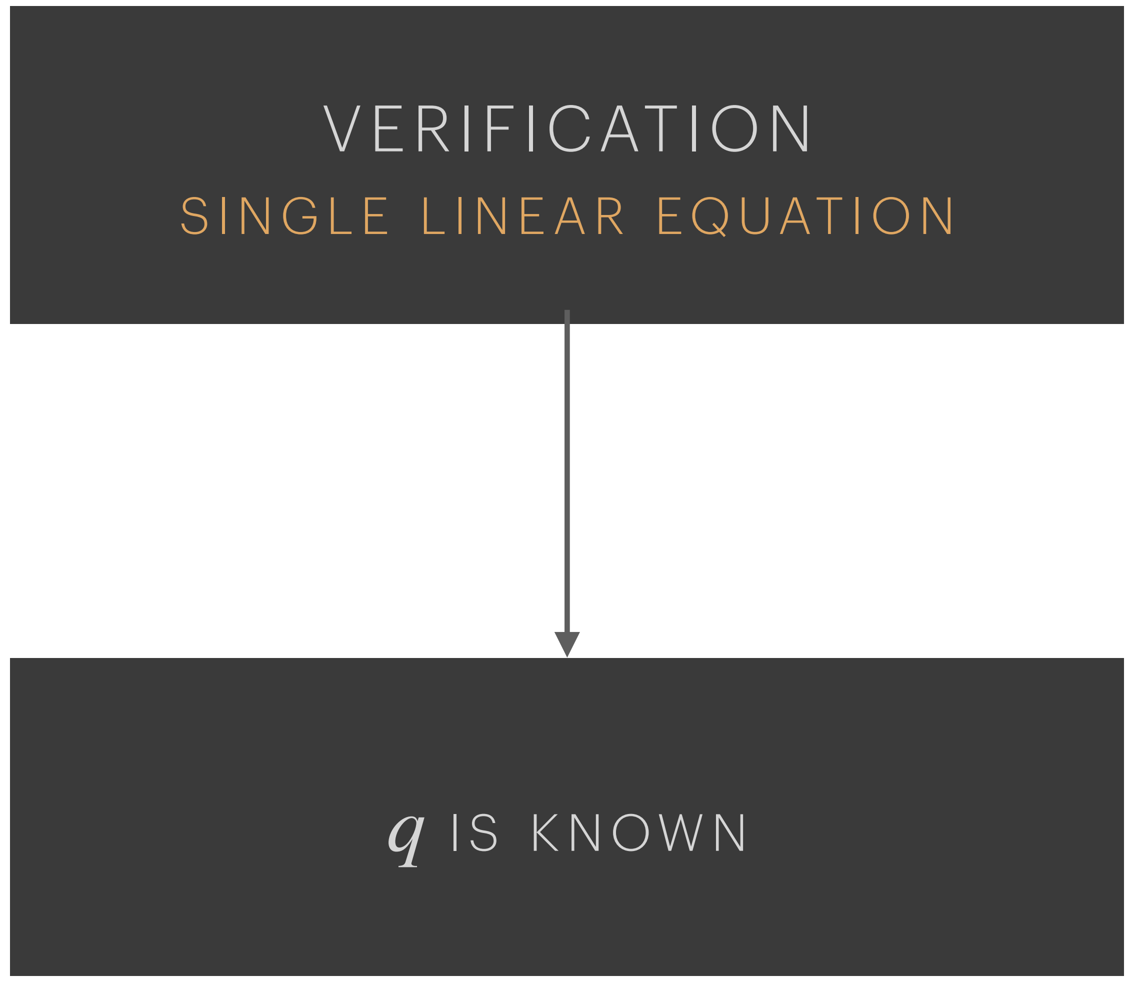
Dimension 2 example



Not a lattice yet…

$q = 7$



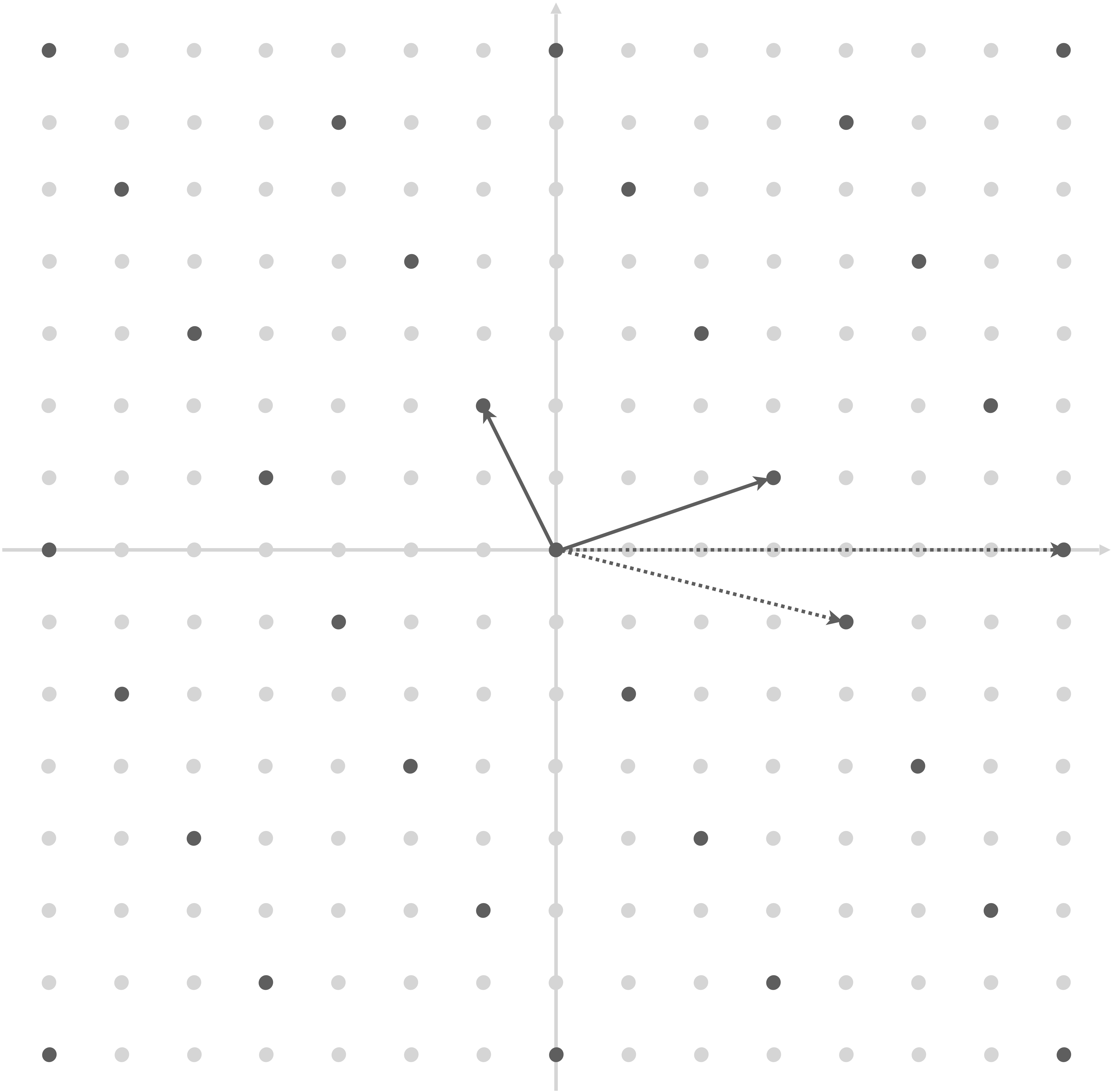


VERIFICATION

SINGLE LINEAR EQUATION

q IS KNOWN

Now it's a lattice



VERIFICATION

SINGLE LINEAR EQUATION

$$-x + 4y = 0 \pmod{q}$$

PRIVATE BASIS

GOOD GEOMETRY

SPECIAL ARITHMETIC

$$\begin{pmatrix} 3 & -1 \\ 1 & 2 \end{pmatrix}$$

PUBLIC BASIS

HERMITE FORM

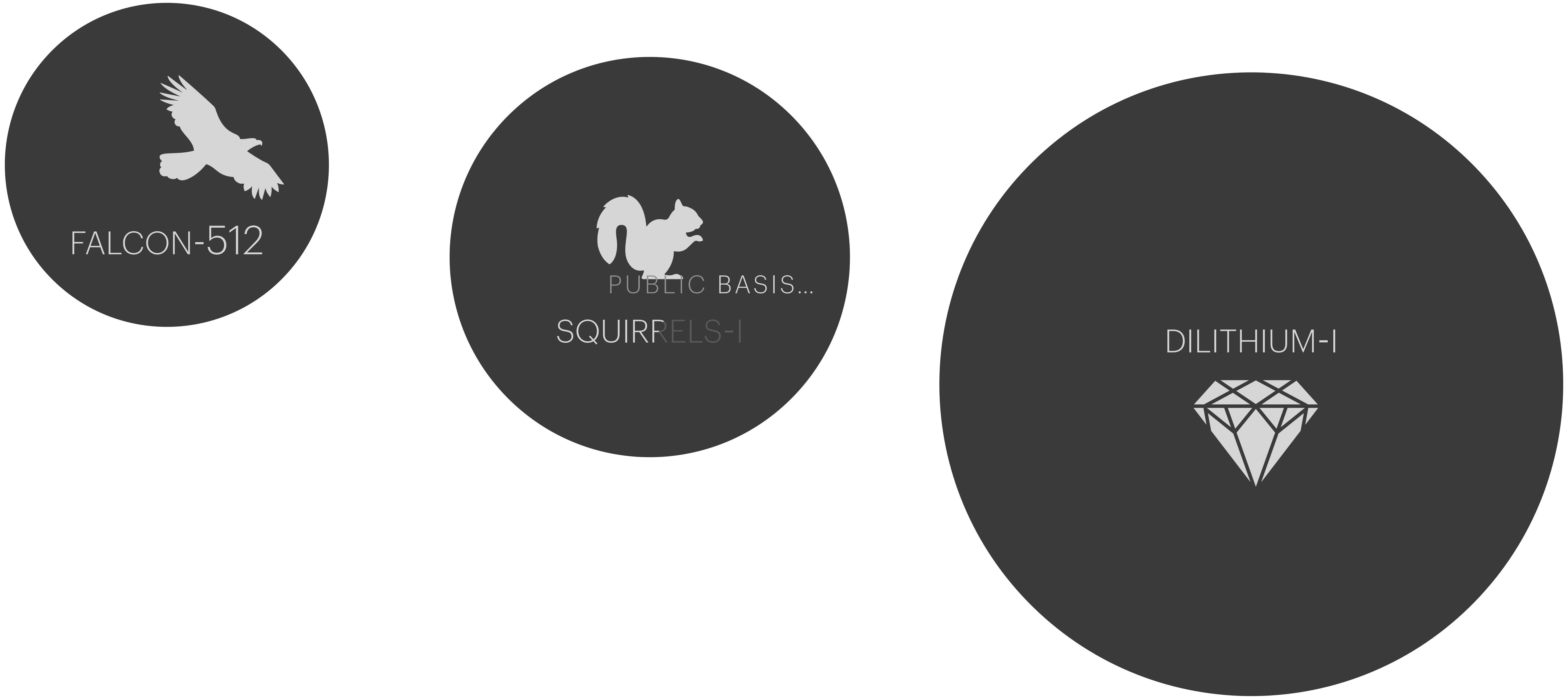
$$\begin{pmatrix} q & 4 \\ 0 & -1 \end{pmatrix}$$

Concrete values (raw!)

	BIT SECURITY CLASSICAL/QUANTUM	SIG-SIZE BYTES	KEY-SIZE BYTES	KEYGEN SECOND	VERIFICATION SIG PER SEC.	SIGN TIME SIG PER SEC.
🐿- I	125/112	**1019**	681780	34	**601**	13099
🐿- II	141/128	**1147**	874576	52	**509**	11871
🐿-III	192/174	**1554**	1629640	127	**266**	6594
🐿-IV	211/192	**1676**	188870	179	**208**	5765
🐿- V	256/232	**2025**	278680	351	**177**	3937

(Performances measured on a Ryzen Pro 7 5850U (16CPU threads at 3GHz)

Size-wise



FALCON-512

PUBLIC BASIS...

SQUIRRELS-I

DILITHIUM-I

SQUIRRELS

+

- UNSTRUCTURED
- FAST + SMALL SIG
- VERY TAILORABLE

-

- HUGE PUBLIC KEY
- HEAVY KEYGEN
- FLOATING POINT