Gottesman-Kitaev-Preskill bosonic error correcting codes: a lattice perspective

Quantum 6, 648 (2022)

Jonathan Conrad, Jens Eisert, Francesco Arzani

Alexander von Humboldt
Stiftung/Foundation

Freie Universität Berlin

Outline

1) Introduction and motivation

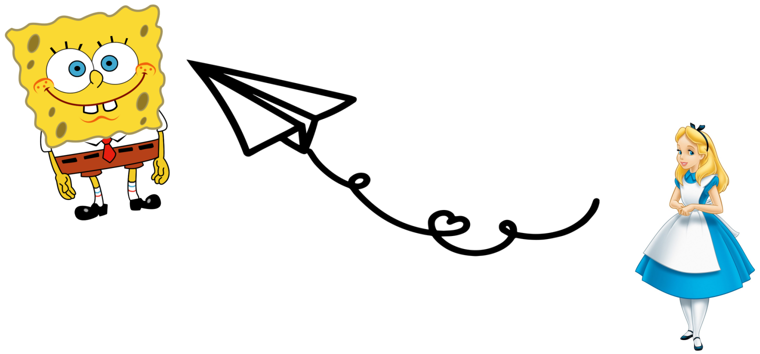
2) Some definitions : Lattices in a nutshell

3) Results : Lattice bases and symplectically equivalent codes

4) Conclusions

Part 1 : Introduction and motivation

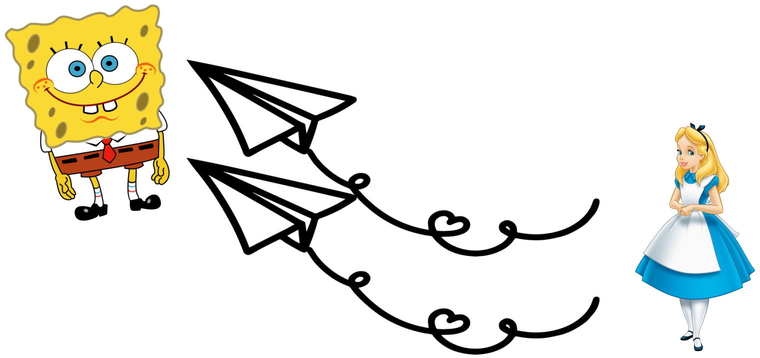
(Quantum) Error correction and harmonic oscillators



Information always encoded in phys. syst.

⇨ Always subject to **noise**

(Quantum) Error correction and harmonic oscillators

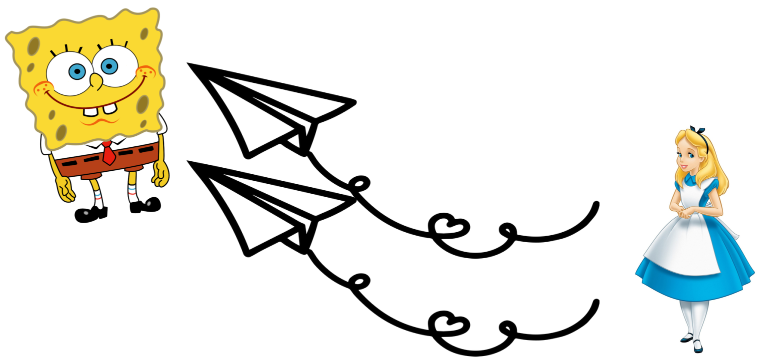


Information always encoded in phys. syst.

⇨ Always subject to **noise**

Error correction: **redundancy**

(Quantum) Error correction and harmonic oscillators



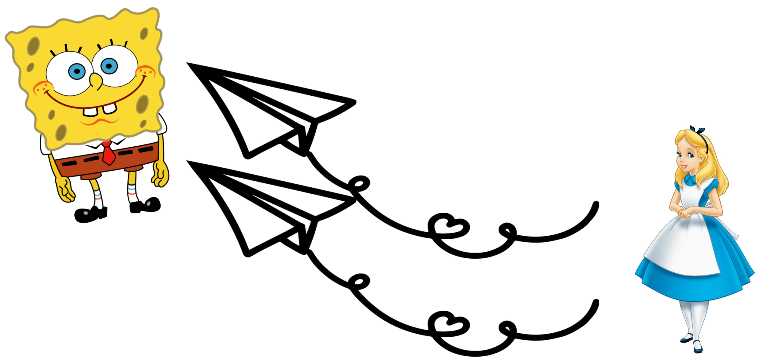
Information always encoded in phys. syst.

Always subject to **noise**

Error correction: **redundancy**

Quantum: *mostly* qubits $\alpha|0\rangle + \beta|1\rangle$

(Quantum) Error correction and harmonic oscillators



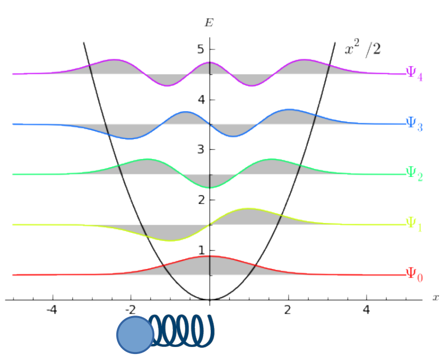
Information always encoded in phys. syst.

⇨ Always subject to **noise**

Error correction: **redundancy**

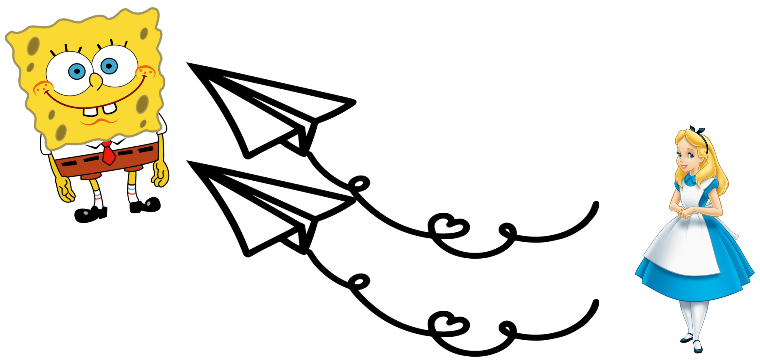
Quantum: *mostly* qubits $\alpha\,|0\rangle + \beta\,|1\rangle$

Bosonic codes: *oscillators*



EM field mode, LC circuit, ...

(Quantum) Error correction and harmonic oscillators



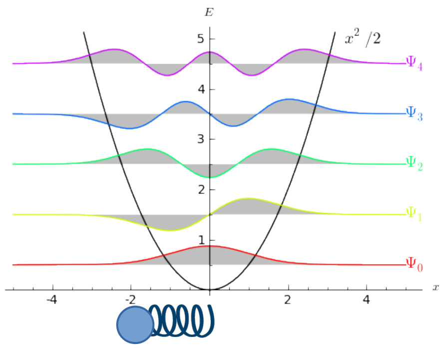
Information always encoded in phys. syst.

⇒ Always subject to **noise**

Error correction: **redundancy**

Quantum: *mostly* qubits $\alpha\left|0\right\rangle + \beta\left|1\right\rangle$

Bosonic codes: <u>*oscillators*</u>



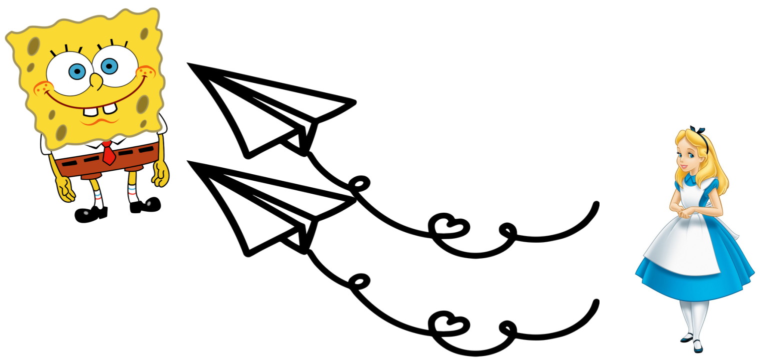
EM field mode, LC circuit, ...

$$\hat{\boldsymbol{x}} = (q_1, \ldots, q_n, p_1, \ldots, p_n)^T$$

$$[\hat{x}_j, \hat{x}_k] = iJ_{jk}$$

$$J = \begin{pmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{pmatrix}$$

(Quantum) Error correction and harmonic oscillators



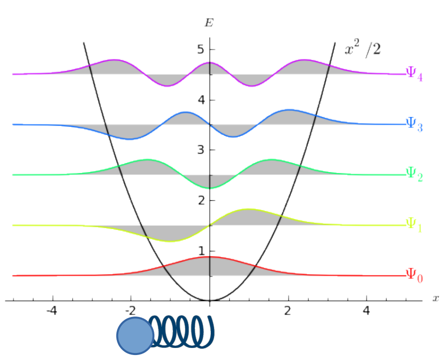
Information always encoded in phys. syst.

Always subject to **noise**

Error correction: **redundancy**

Quantum: *mostly* qubits $\alpha\,|0\rangle + \beta\,|1\rangle$

Bosonic codes: <u>*oscillators*</u>



EM field mode, LC circuit, ...

$$\hat{\boldsymbol{x}} = (q_1, \ldots, q_n, p_1, \ldots, p_n)^T$$

$$[\hat{x}_j, \hat{x}_k] = iJ_{jk}$$

$$J = \begin{pmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{pmatrix}$$

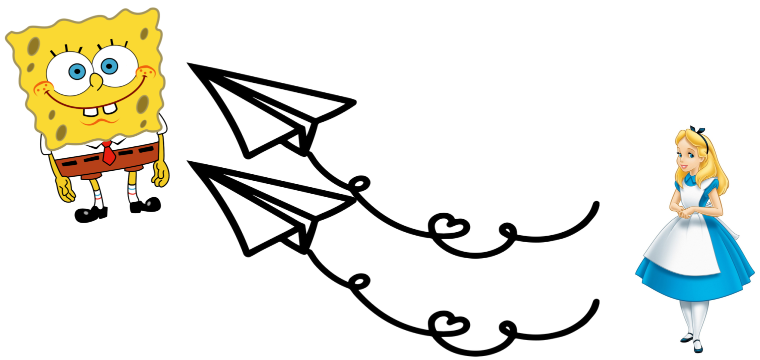
We have *continuous* variables

$$|\psi\rangle = \int \mathrm{d}q\,\psi(q)\,|q\rangle$$

Infinite dimensional space!

How to exploit?

(Quantum) Error correction and harmonic oscillators



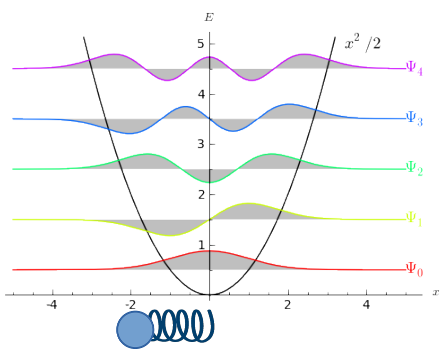
Information always encoded in phys. syst.

Always subject to **noise**

Error correction: **redundancy**

Quantum: *mostly* qubits $\alpha |0\rangle + \beta |1\rangle$

Bosonic codes: <u>*oscillators*</u>



EM field mode, LC circuit, ...

$$\hat{\boldsymbol{x}} = (q_1, \ldots, q_n, p_1, \ldots, p_n)^T$$

$$[\hat{x}_j, \hat{x}_k] = iJ_{jk}$$

$$J = \begin{pmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{pmatrix}$$

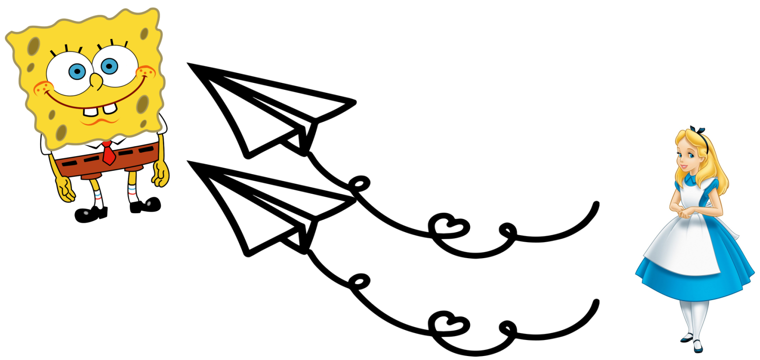
We have *continuous* variables

$$|\psi\rangle = \int \mathrm{d}q \psi(q) |q\rangle$$

Infinite dimensional space!
How to exploit?

Symmetries!

(Quantum) Error correction and harmonic oscillators



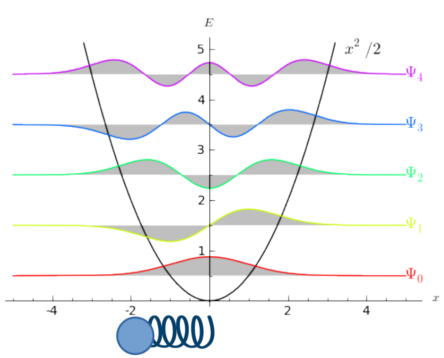
Information always encoded in phys. syst.

⇨ Always subject to **noise**

Error correction: **redundancy**

Quantum: *mostly* qubits $\alpha |0\rangle + \beta |1\rangle$

Bosonic codes: <u>*oscillators*</u>



EM field mode, LC circuit, ...

$$\hat{\boldsymbol{x}} = (q_1, \ldots, q_n, p_1, \ldots, p_n)^T$$

$$[\hat{x}_j, \hat{x}_k] = iJ_{jk}$$

$$J = \begin{pmatrix} \mathbf{0} & \mathbb{I} \\ -\mathbb{I} & \mathbf{0} \end{pmatrix}$$

We have *continuous* variables

$$|\psi\rangle = \int \mathrm{d}q \psi(q) |q\rangle$$

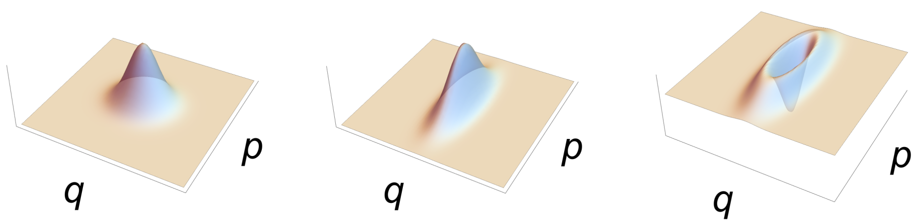
Infinite dimensional space!

How to exploit?

Symmetries!

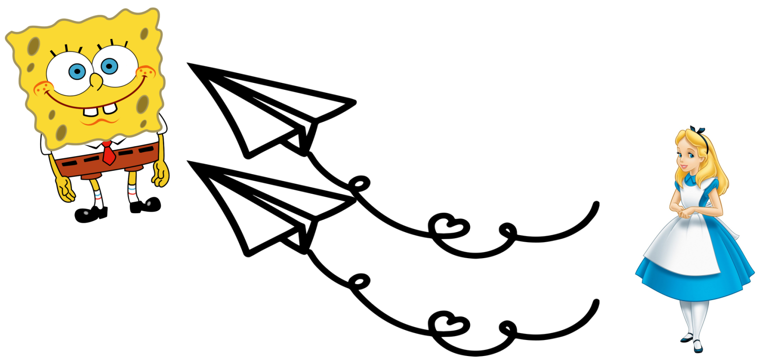
In phase space:

Wigner Function



Quasi-probability distribution

(Quantum) Error correction and harmonic oscillators



Information always encoded in phys. syst.

⇒ Always subject to **noise**

Error correction: **redundancy**

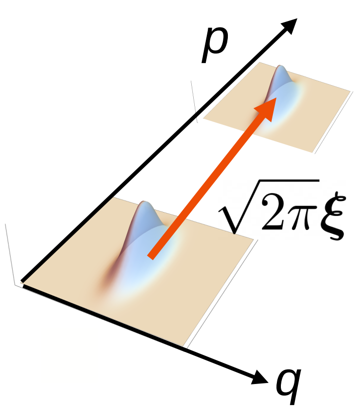
We have *continuous* variables

$$|\psi\rangle = \int \mathrm{d}q \psi(q) |q\rangle$$

Infinite dimensional space!

How to exploit?

Symmetries!



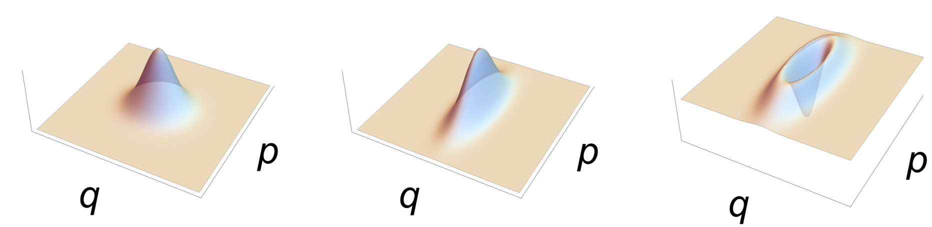
Displacements :

$$D^\dagger(\boldsymbol{\xi})\hat{\boldsymbol{x}}D(\boldsymbol{\xi}) = \hat{\boldsymbol{x}} + \sqrt{2\pi}\boldsymbol{\xi}$$

$$D(\boldsymbol{\xi})D(\boldsymbol{\eta}) = e^{-i2\pi\boldsymbol{\xi}^T J\boldsymbol{\eta}}D(\boldsymbol{\eta})D(\boldsymbol{\xi})$$

In phase space:

Wigner Function



Quasi-probability distribution

Encoding qubits on a lattice

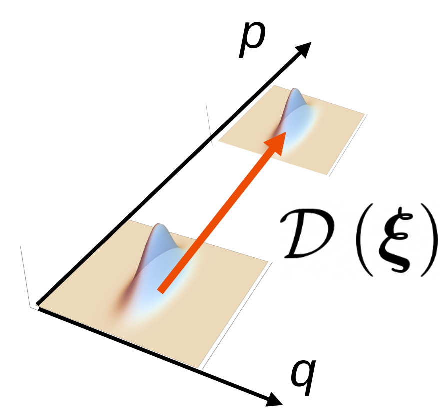
“Logical subspace”: finite photon number, odd/even photon number, or...**translation symmetries**!

Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**



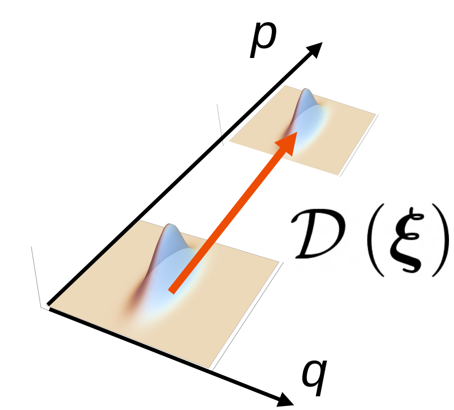
Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**

$$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \ \ldots, \ \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle$$ ⇨ Code: $$\mathcal{D}(\boldsymbol{\xi}_j) |\psi\rangle = |\psi\rangle$$



Encoding qubits on a lattice

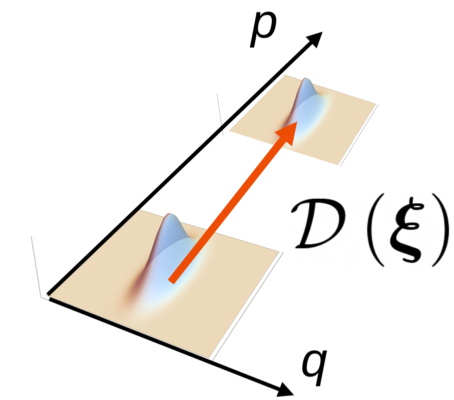
"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**

$$\mathcal{S} = \langle \mathcal{D}\left(\boldsymbol{\xi}_1\right),\ \ldots,\ \mathcal{D}\left(\boldsymbol{\xi}_{2n}\right)\rangle$$ ⇨ Code: $$\mathcal{D}\left(\boldsymbol{\xi}_j\right)|\psi\rangle = |\psi\rangle$$

$$\boxed{\mathcal{D}\left(\boldsymbol{\xi}_j\right)\mathcal{D}\left(\boldsymbol{\xi}_k\right) = e^{if(j,k)}\mathcal{D}\left(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k\right)}$$



Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

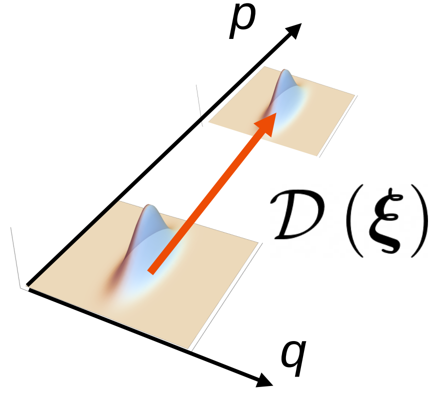
Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**

$$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \; \ldots, \; \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle$$ ⇨ Code: $\mathcal{D}(\boldsymbol{\xi}_j)\,|\psi\rangle = |\psi\rangle$

$$\boxed{\mathcal{D}(\boldsymbol{\xi}_j)\,\mathcal{D}(\boldsymbol{\xi}_k) = e^{if(j,k)}\mathcal{D}(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k)}$$

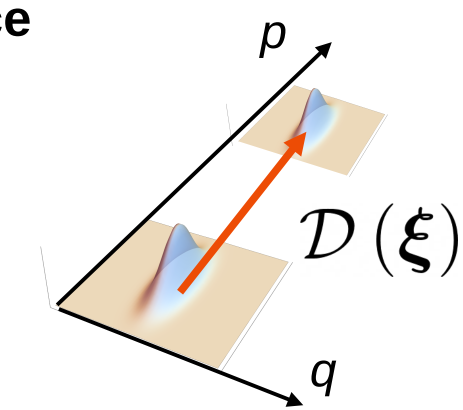
$$\boxed{\mathcal{S} \cong \mathcal{L} = \left\{ \textstyle\sum_j z_j \boldsymbol{\xi}_j \; : \; z_j \in \mathbb{Z} \right\}}$$



Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**



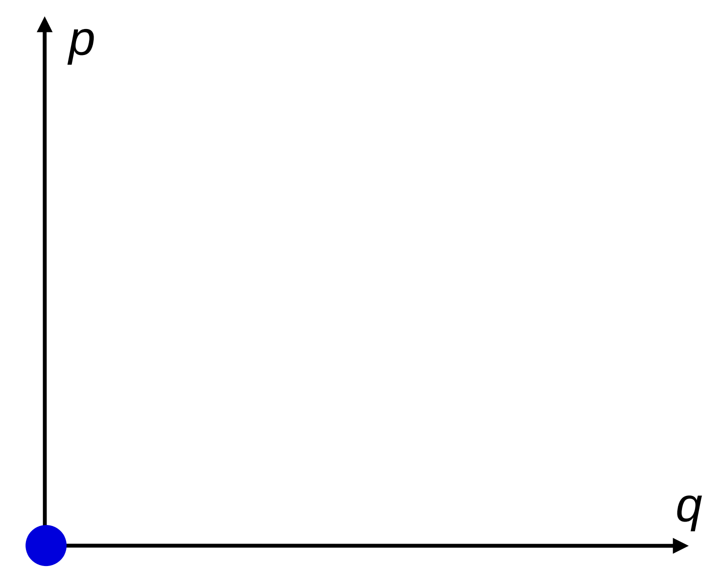
Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**

$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \ \ldots, \ \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle$ ⇒ Code: $\mathcal{D}(\boldsymbol{\xi}_j) |\psi\rangle = |\psi\rangle$

$$\boxed{\mathcal{D}(\boldsymbol{\xi}_j)\, \mathcal{D}(\boldsymbol{\xi}_k) = e^{i f(j,k)} \mathcal{D}(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \{ \textstyle\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z} \}}$$

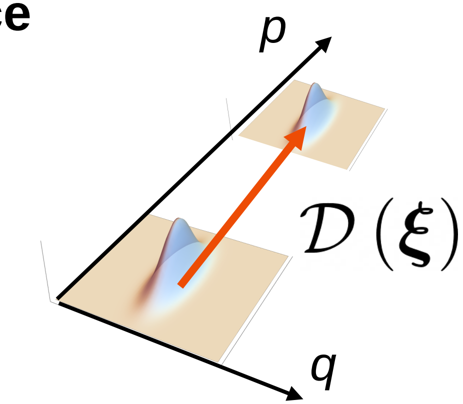
Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$



Encoding qubits on a lattice

“Logical subspace”: finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**



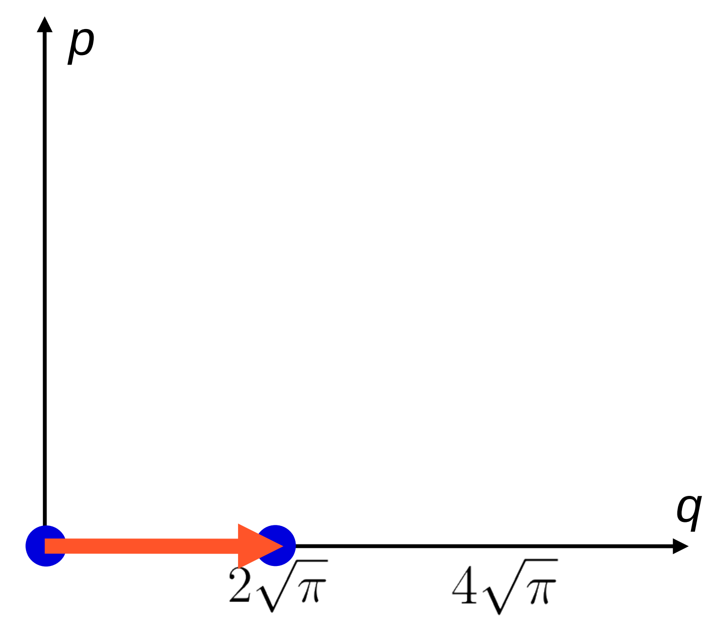
Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**

$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \ \ldots, \ \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle$ ⇒ Code: $\mathcal{D}(\boldsymbol{\xi}_j)\,|\psi\rangle = |\psi\rangle$

$$\boxed{\mathcal{D}(\boldsymbol{\xi}_j)\,\mathcal{D}(\boldsymbol{\xi}_k) = e^{if(j,k)}\mathcal{D}(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \{\textstyle\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z}\}}$$

Ex : 1 qubit in 1 mode → $\boxed{\boldsymbol{\xi}_q}, \ \boldsymbol{\xi}_p$



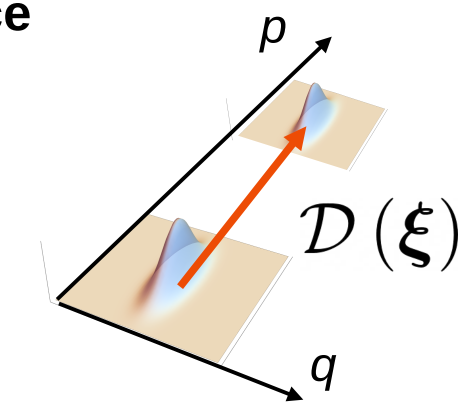
Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**

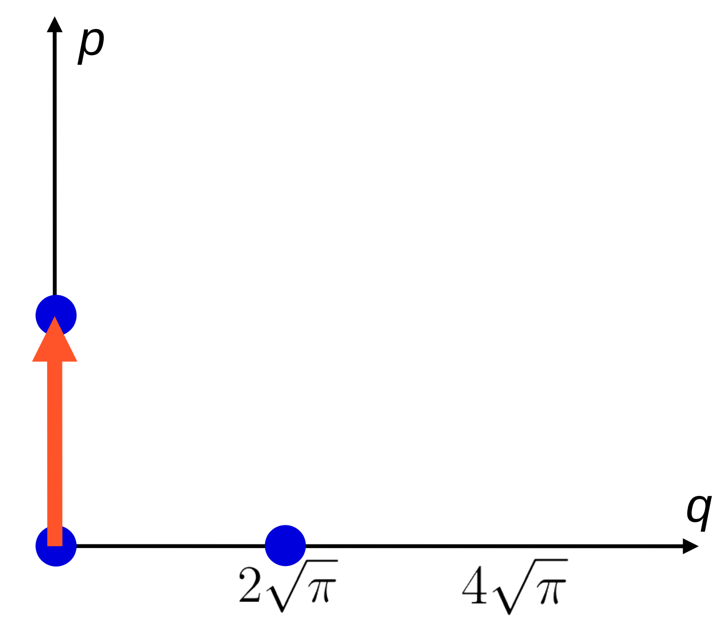
$$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \ \ldots, \ \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle \implies \text{Code: } \mathcal{D}(\boldsymbol{\xi}_j)\,|\psi\rangle = |\psi\rangle$$



$$\boxed{\mathcal{D}(\boldsymbol{\xi}_j)\,\mathcal{D}(\boldsymbol{\xi}_k) = e^{if(j,k)}\mathcal{D}(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \{\textstyle\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z}\}}$$

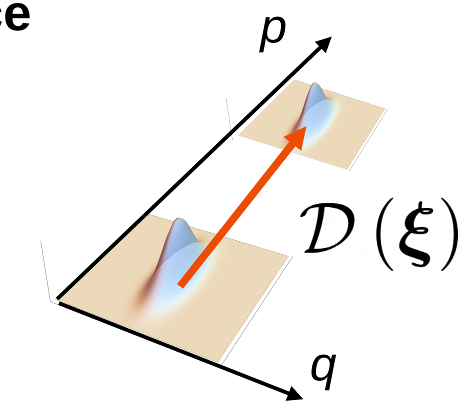
Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \boxed{\boldsymbol{\xi}_p}$



Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**



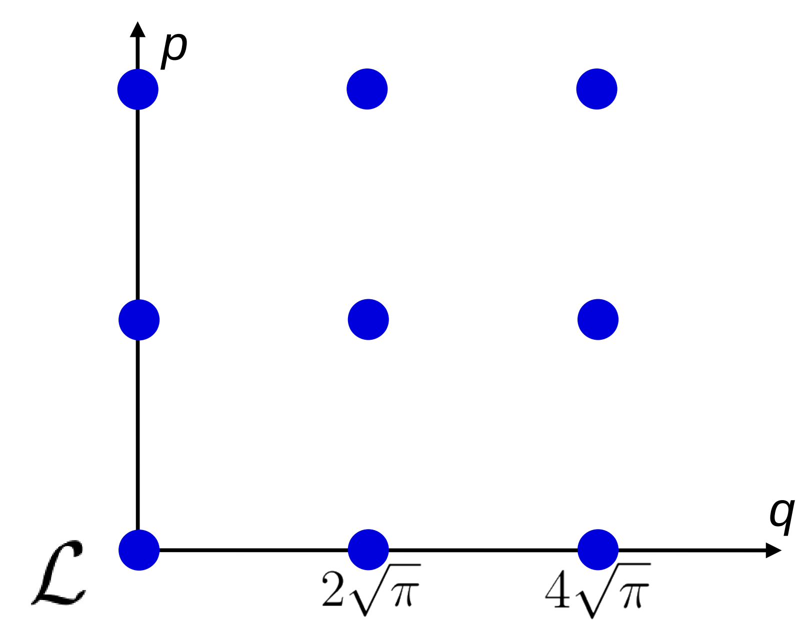
Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**

$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \ \ldots, \ \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle$ ⇒ Code: $\mathcal{D}(\boldsymbol{\xi}_j)\,|\psi\rangle = |\psi\rangle$

$$\mathcal{D}(\boldsymbol{\xi}_j)\,\mathcal{D}(\boldsymbol{\xi}_k) = e^{if(j,k)}\mathcal{D}(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k)$$

$$\mathcal{S} \cong \mathcal{L} = \left\{\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z}\right\}$$

Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$

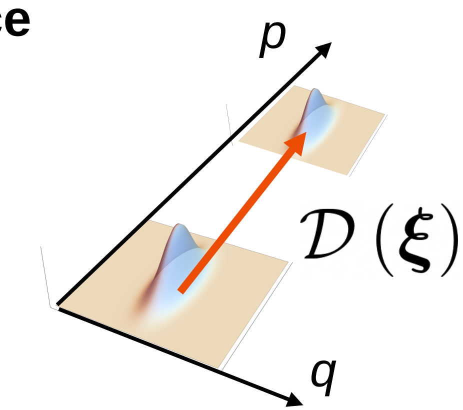


Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**



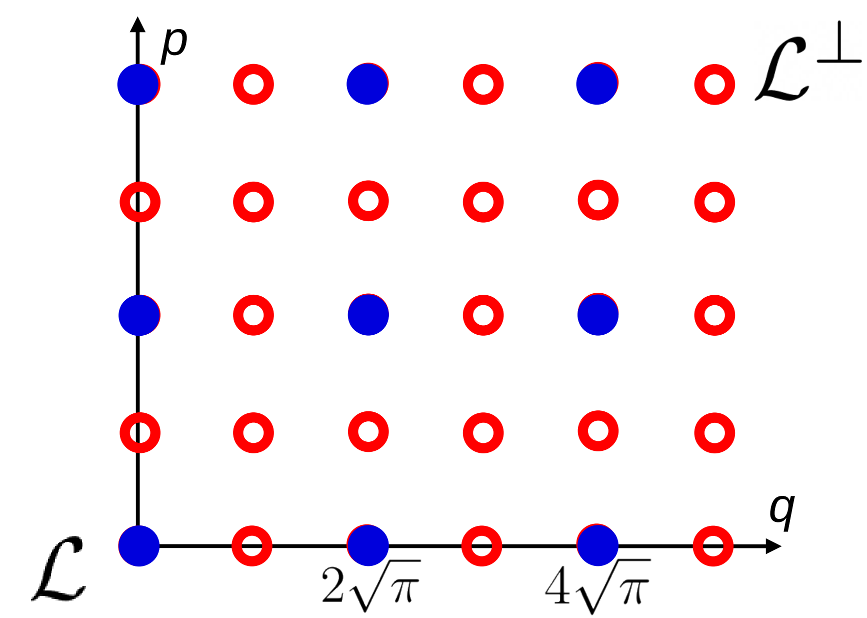
$\mathcal{S} = \langle \mathcal{D}(\boldsymbol{\xi}_1), \ \ldots, \ \mathcal{D}(\boldsymbol{\xi}_{2n}) \rangle$ ⇒ Code: $\mathcal{D}\left(\boldsymbol{\xi}_j\right) |\psi\rangle = |\psi\rangle$

$$\boxed{\mathcal{D}\left(\boldsymbol{\xi}_j\right) \mathcal{D}\left(\boldsymbol{\xi}_k\right) = e^{if(j,k)} \mathcal{D}\left(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k\right)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \left\{ \sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z} \right\}}$$

$\mathcal{L}^{\perp}$ → Logical Pauli operators

Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$

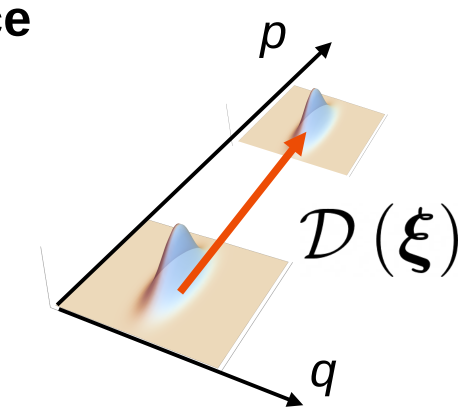


Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**



$$\mathcal{S} = \langle \mathcal{D}\left(\boldsymbol{\xi}_1\right), \ \ldots, \ \mathcal{D}\left(\boldsymbol{\xi}_{2n}\right) \rangle \Rightarrow \text{Code: } \mathcal{D}\left(\boldsymbol{\xi}_j\right) |\psi\rangle = |\psi\rangle$$

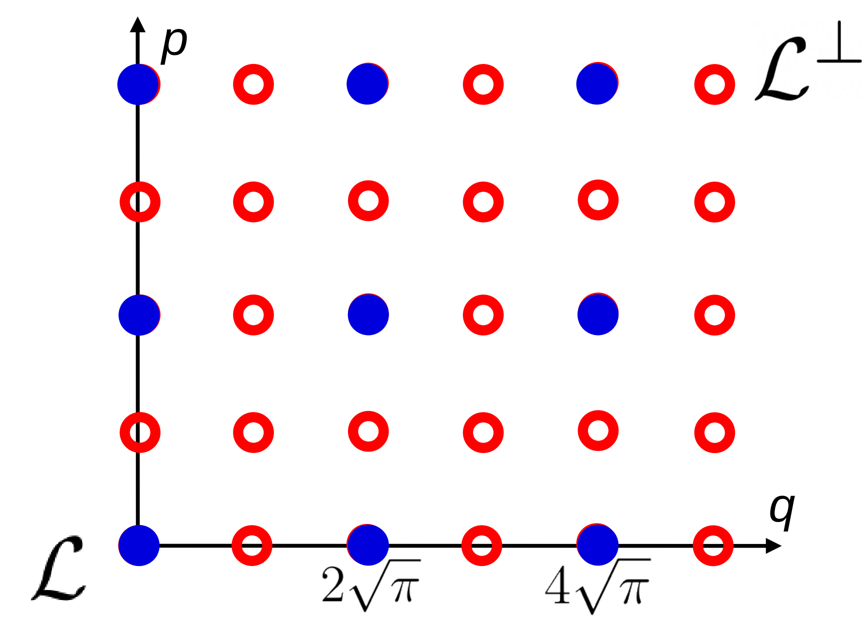
« lives » on $\mathcal{L}^{\perp}$

$$\boxed{\mathcal{D}\left(\boldsymbol{\xi}_j\right) \mathcal{D}\left(\boldsymbol{\xi}_k\right) = e^{if(j,k)} \mathcal{D}\left(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k\right)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \left\{ \sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z} \right\}}$$

$\mathcal{L}^{\perp}$ → Logical Pauli operators

Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$

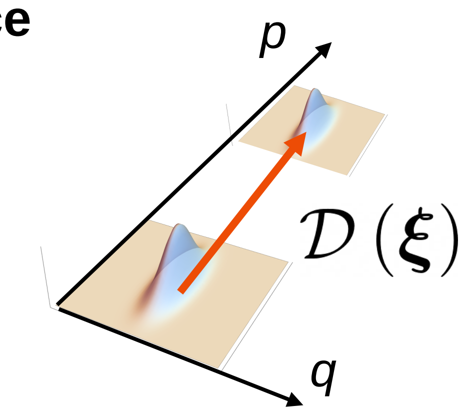


Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**



$$\mathcal{S} = \langle \mathcal{D}\left(\boldsymbol{\xi}_1\right), \ \dots, \ \mathcal{D}\left(\boldsymbol{\xi}_{2n}\right) \rangle$$ ⇒ Code: $\mathcal{D}\left(\boldsymbol{\xi}_j\right) |\psi\rangle = |\psi\rangle$

« lives » on $\mathcal{L}^{\perp}$

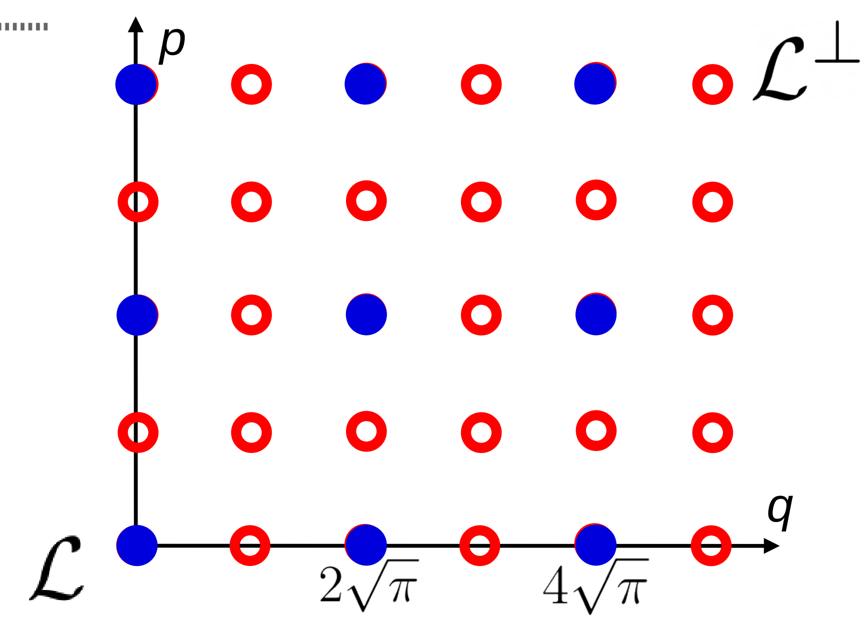
$$\boxed{\mathcal{D}\left(\boldsymbol{\xi}_j\right)\mathcal{D}\left(\boldsymbol{\xi}_k\right) = e^{if(j,k)}\mathcal{D}\left(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k\right)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \{\textstyle\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z}\}}$$

$\mathcal{L}^{\perp}$ → Logical Pauli operators

Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$

- Good protection against common noise processes
  Albert et al, PRA 97 (2018)



Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**

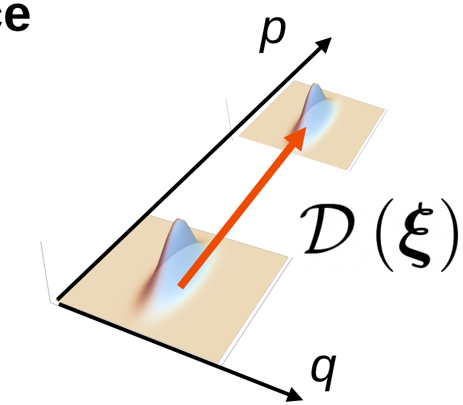
$$\mathcal{S} = \langle \mathcal{D}(\xi_1), \; \ldots, \; \mathcal{D}(\xi_{2n}) \rangle \Rightarrow \text{Code: } \mathcal{D}(\xi_j) \left|\psi\right\rangle = \left|\psi\right\rangle$$

« lives » on $\mathcal{L}^\perp$

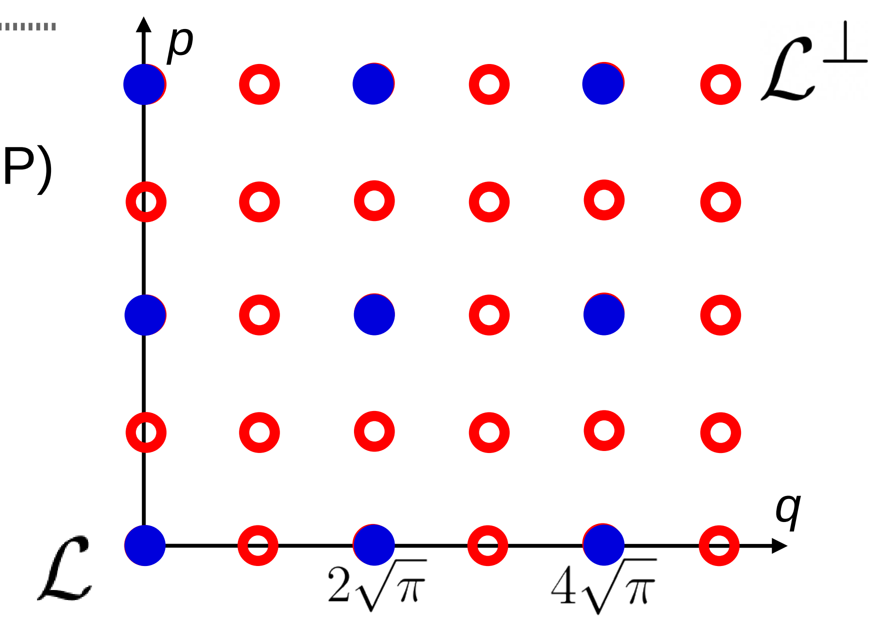
$$\boxed{\mathcal{D}(\xi_j)\,\mathcal{D}(\xi_k) = e^{if(j,k)}\mathcal{D}(\xi_j + \xi_k)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \{\textstyle\sum_j z_j \xi_j \; : \; z_j \in \mathbb{Z}\}}$$

$\mathcal{L}^\perp$ → Logical Pauli operators



Ex : 1 qubit in 1 mode → $\xi_q, \; \xi_p$



- Good protection against common noise processes
  Albert et al, PRA 97 (2018)
- Logical Clifford = Gaussian operations ("easy" - good for EC & QIP)
  Gottesman, Kitaev, Preskill PRA 64 (2001)

Encoding qubits on a lattice

“Logical subspace”: finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

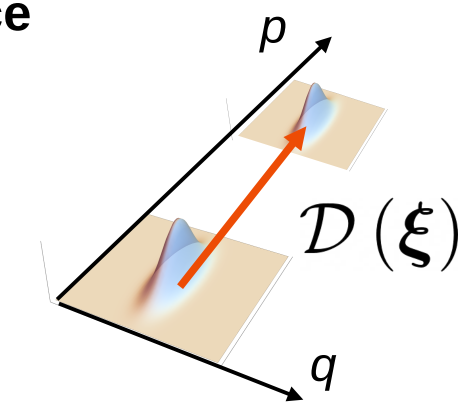
Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**

$\mathcal{S} = \langle \mathcal{D}(\xi_1), \ldots, \mathcal{D}(\xi_{2n}) \rangle$ ⇒ Code: $\mathcal{D}(\xi_j)|\psi\rangle = |\psi\rangle$

$$\boxed{\mathcal{D}(\xi_j)\mathcal{D}(\xi_k) = e^{if(j,k)}\mathcal{D}(\xi_j + \xi_k)}$$

« lives » on $\mathcal{L}^\perp$

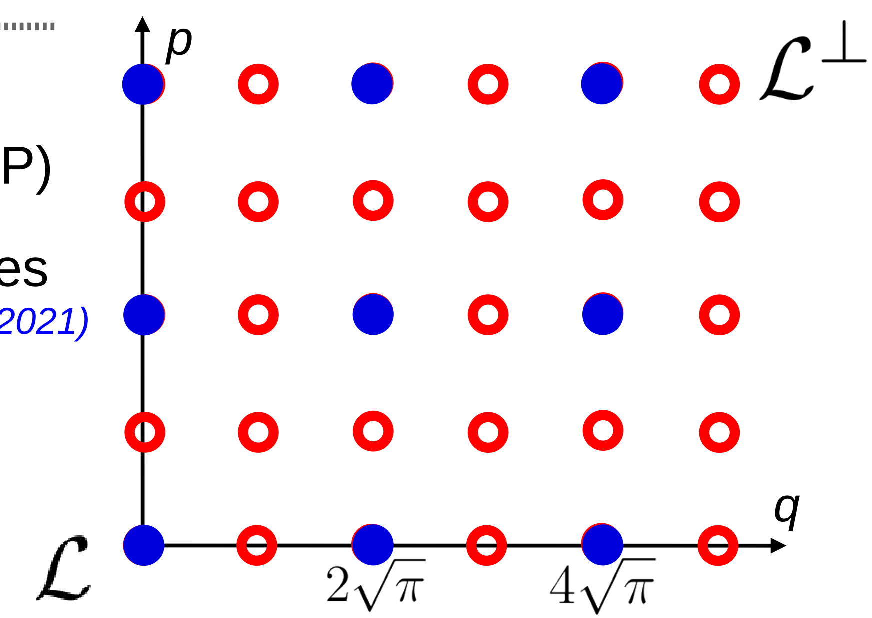
$$\boxed{\mathcal{S} \cong \mathcal{L} = \{\textstyle\sum_j z_j \xi_j \ : \ z_j \in \mathbb{Z}\}}$$



$\mathcal{L}^\perp$ → Logical Pauli operators

Ex : 1 qubit in 1 mode → $\xi_q, \ \xi_p$

- Good protection against common noise processes
  Albert et al, PRA 97 (2018)
- Logical Clifford = Gaussian operations (“easy” - good for EC & QIP)
  Gottesman, Kitaev, Preskill PRA 64 (2001)
- Can be used as effective qubits and combined with stabilizer codes
  Vuillot et al, PRA 99 (2019) *Noh&Chamberland PRA 101 (2020)* *Bourassa et al, Quantum 5 (2021)*

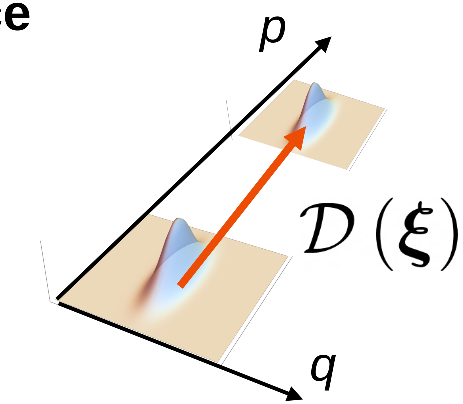


Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇨ **GKP codes**



$\mathcal{S} = \langle \mathcal{D}\left(\boldsymbol{\xi}_1\right), \ \ldots, \ \mathcal{D}\left(\boldsymbol{\xi}_{2n}\right) \rangle$ ⇨ Code: $\mathcal{D}\left(\boldsymbol{\xi}_j\right)|\psi\rangle = |\psi\rangle$

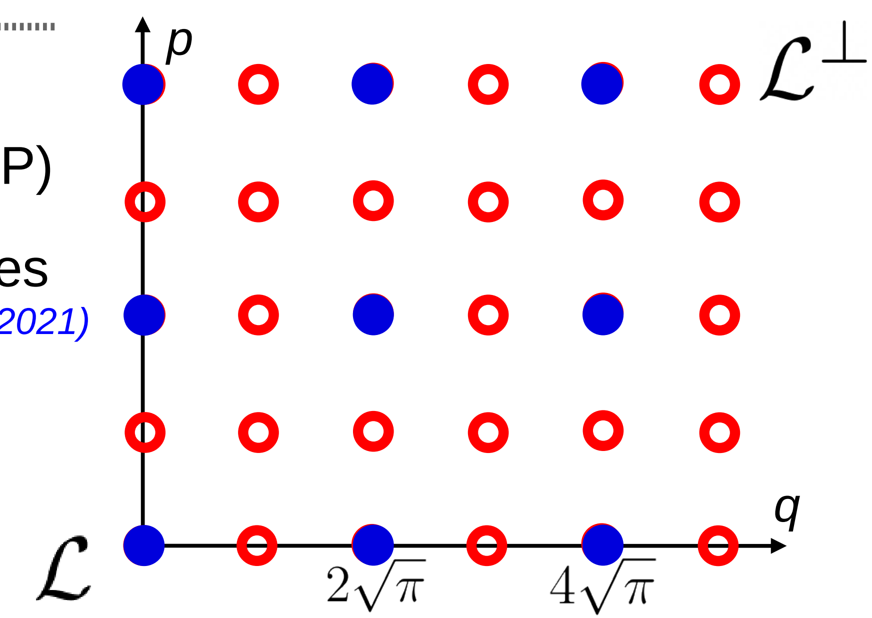
$$\boxed{\mathcal{D}\left(\boldsymbol{\xi}_j\right)\mathcal{D}\left(\boldsymbol{\xi}_k\right) = e^{if(j,k)}\mathcal{D}\left(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k\right)}$$

« lives » on $\mathcal{L}^{\perp}$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \{\textstyle\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z}\}}$$

$\mathcal{L}^{\perp}$ → Logical Pauli operators

Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$



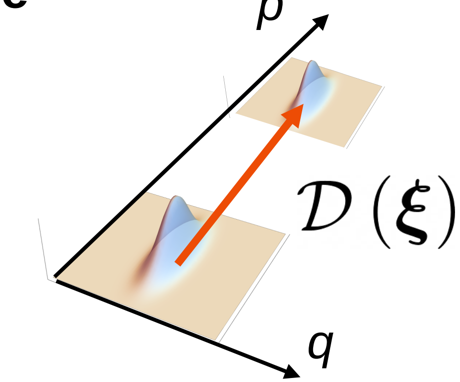
- Good protection against common noise processes
  Albert et al, PRA 97 (2018)
- Logical Clifford = Gaussian operations ("easy" - good for EC & QIP)
  Gottesman, Kitaev, Preskill PRA 64 (2001)
- Can be used as effective qubits and combined with stabilizer codes
  Vuillot et al, PRA 99 (2019) *Noh&Chamberland PRA 101 (2020)* *Bourassa et al, Quantum 5 (2021)*
- Can protect CV systems
  Noh et al, PRL 125 (2020)

Encoding qubits on a lattice

"Logical subspace": finite photon number, odd/even photon number, or...**translation symmetries**!

Grid codes: **stabilized** by (commuting) **displacement** operators → underlying **lattice**

Gottesman, Kitaev, Preskill PRA 64 (2001) ⇒ **GKP codes**



$$\mathcal{S} = \langle \mathcal{D}\left(\boldsymbol{\xi}_1\right), \ \ldots, \ \mathcal{D}\left(\boldsymbol{\xi}_{2n}\right) \rangle \Rightarrow \text{Code: } \mathcal{D}\left(\boldsymbol{\xi}_j\right)|\psi\rangle = |\psi\rangle$$

« lives » on $\mathcal{L}^{\perp}$

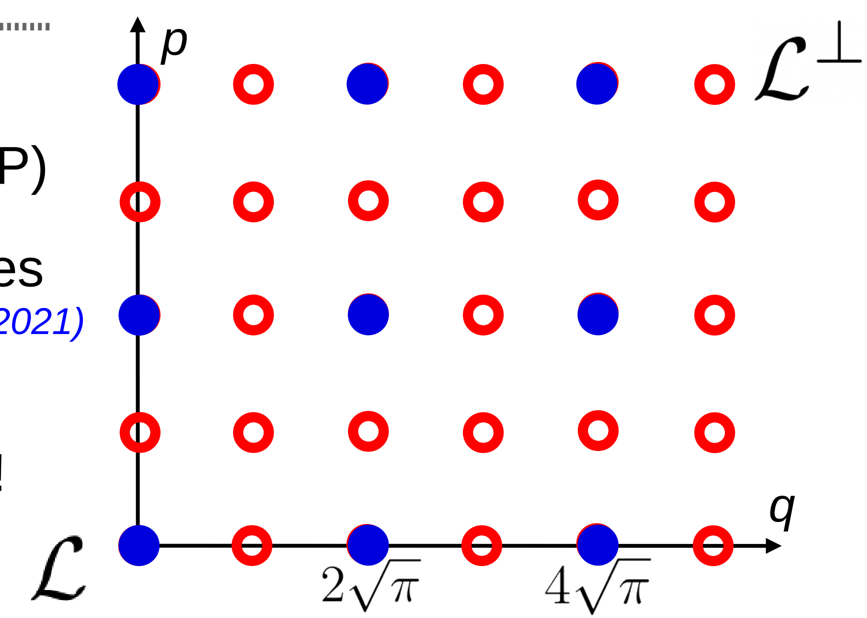
$$\boxed{\mathcal{D}\left(\boldsymbol{\xi}_j\right)\mathcal{D}\left(\boldsymbol{\xi}_k\right) = e^{if(j,k)}\mathcal{D}\left(\boldsymbol{\xi}_j + \boldsymbol{\xi}_k\right)}$$

$$\boxed{\mathcal{S} \cong \mathcal{L} = \left\{\textstyle\sum_j z_j \boldsymbol{\xi}_j \ : \ z_j \in \mathbb{Z}\right\}}$$

$\mathcal{L}^{\perp}$ → Logical Pauli operators

Ex : 1 qubit in 1 mode → $\boldsymbol{\xi}_q, \ \boldsymbol{\xi}_p$

- Good protection against common noise processes
  Albert et al, PRA 97 (2018)
- Logical Clifford = Gaussian operations ("easy" - good for EC & QIP)
  Gottesman, Kitaev, Preskill PRA 64 (2001)
- Can be used as effective qubits and combined with stabilizer codes
  Vuillot et al, PRA 99 (2019) *Noh&Chamberland PRA 101 (2020)* *Bourassa et al, Quantum 5 (2021)*
- Can protect CV systems
  Noh et al, PRL 125 (2020)
- Logical states thought hard to realize, now there are experiments!
  Flühmann et al, Nature 566 (2019) *Campagne-Ibarcq et al, Nature 584 (2020)*



The lattice point of view

The lattice point of view

For exponential noise suppression: more oscillators

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ “lattice picture” only for individual oscillators,
not for **whole code**

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ “lattice picture” only for individual oscillators,
not for **whole code**

Q: Can lattice properties be exploited more?

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ “lattice picture” only for individual oscillators,
not for **whole code**

Q: Can lattice properties be exploited more?

upshot:
lattices are very well studied!

J. Conway and N. Sloane.
Sphere packings, lattices and groups, volume 290. 1988

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ “lattice picture” only for individual oscillators,
not for **whole code**

Q: Can lattice properties be exploited more?

upshot:
lattices are very well studied!

J. Conway and N. Sloane.
Sphere packings, lattices and groups, volume 290. 1988

…but not so much for GKP!

Gottesman, Kitaev, Preskill PRA 64 (2001)

Harrington, Preskill PRA 64 (2001)

Hänggli, Heinze, König, PRA 102 (2020)

Hänggli, König, IEEE TIT 68(2) (2021)

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ "lattice picture" only for individual oscillators,
not for **whole code**

Q: Can lattice properties be exploited more?

upshot:
lattices are very well studied!

J. Conway and N. Sloane.
Sphere packings, lattices and groups, volume 290. 1988

…but not so much for GKP!

Gottesman, Kitaev, Preskill PRA 64 (2001)

Harrington, Preskill PRA 64 (2001)

Hänggli, Heinze, König, PRA 102 (2020)

Hänggli, König, IEEE TIT 68(2) (2021)

Since our work came out :

Schmidt, van Loock, PRA 105 (2022)

Royer, Singh, Girvin, PRX Quantum 105 (2022)

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ “lattice picture” only for individual oscillators,
not for **whole code**

Q: Can lattice properties be exploited more? Yes!

upshot:
lattices are very well studied!

J. Conway and N. Sloane.
Sphere packings, lattices and groups, volume 290. 1988

…but not so much for GKP!

Gottesman, Kitaev, Preskill PRA 64 (2001)

Harrington, Preskill PRA 64 (2001)

Hänggli, Heinze, König, PRA 102 (2020)

Hänggli, König, IEEE TIT 68(2) (2021)

Since our work came out :

Schmidt, van Loock, PRA 105 (2022)

Royer, Singh, Girvin, PRX Quantum 105 (2022)

Quantum 6, 648 (2022)

Results

1. Code properties from lattice bases
2. Symplectic operations
3. Distance bounds for GKP codes
4. Decoding problem and Θ functions
5. GKP codes beyond concatenation

The lattice point of view

For exponential noise suppression: more oscillators

Up to now: *concatenation* → regard as effective qubits, add qubit-level code
→ "lattice picture" only for individual oscillators,
not for **whole code**

Q: Can lattice properties be exploited more? Yes!

upshot:
lattices are very well studied!

J. Conway and N. Sloane.
Sphere packings, lattices and groups, volume 290. 1988

…but not so much for GKP!

Gottesman, Kitaev, Preskill PRA 64 (2001)

Harrington, Preskill PRA 64 (2001)

Hänggli, Heinze, König, PRA 102 (2020)

Hänggli, König, IEEE TIT 68(2) (2021)

Since our work came out :

Schmidt, van Loock, PRA 105 (2022)

Royer, Singh, Girvin, PRX Quantum 105 (2022)

Quantum 6, 648 (2022)

Results

1. **Code properties from lattice bases**
2. **Symplectic operations**
3. Distance bounds for GKP codes
4. Decoding problem and Θ functions
5. GKP codes beyond concatenation

Part 2 : Definitions

Lattices and stabilizer group

Start from

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle \quad \leftrightarrow \quad M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

$$\Rightarrow \boxed{\mathcal{S} \simeq \mathcal{L} = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{2n} \middle| \boldsymbol{\xi}^T = \boldsymbol{a}^T M,\ \boldsymbol{a} \in \mathbb{Z}^{2n} \right\}}$$

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and stabilizer group

Start from

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle \quad \leftrightarrow \quad M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

$$\Rightarrow \boxed{\mathcal{S} \simeq \mathcal{L} = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{2n} \middle| \boldsymbol{\xi}^T = \boldsymbol{a}^T M,\ \boldsymbol{a} \in \mathbb{Z}^{2n} \right\}}$$

Commutation $\leftrightarrow$ Integer symplectic product $\left[D(\boldsymbol{\xi}_j), D(\boldsymbol{\xi}_j) \right] = 0 \Leftrightarrow \boldsymbol{\xi}_j^T J \boldsymbol{\xi}_k \in \mathbb{Z}$

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and stabilizer group

Start from

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle \quad \leftrightarrow \quad M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

$$\Rightarrow \boxed{\mathcal{S} \simeq \mathcal{L} = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{2n} | \boldsymbol{\xi}^T = \boldsymbol{a}^T M,\ \boldsymbol{a} \in \mathbb{Z}^{2n} \right\}}$$

Commutation $\leftrightarrow$ Integer symplectic product $\left[D(\boldsymbol{\xi}_j), D(\boldsymbol{\xi}_j) \right] = 0 \Leftrightarrow \boldsymbol{\xi}_j^T J \boldsymbol{\xi}_k \in \mathbb{Z}$

$$\boxed{A_{jk}} = \left(M J M^T \right)_{jk} \in \mathbb{Z} \quad \Rightarrow \quad \text{Symplectically integral lattices}$$

Symplectic Gram matrix

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and stabilizer group

Start from

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle \quad \leftrightarrow \quad M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

$$\Rightarrow \boxed{\mathcal{S} \simeq \mathcal{L} = \left\{ \boldsymbol{\xi} \in \mathbb{R}^{2n} \middle| \boldsymbol{\xi}^T = \boldsymbol{a}^T M, \ \boldsymbol{a} \in \mathbb{Z}^{2n} \right\}}$$

Commutation $\leftrightarrow$ Integer symplectic product $\left[D(\boldsymbol{\xi}_j), D(\boldsymbol{\xi}_j)\right] = 0 \Leftrightarrow \boldsymbol{\xi}_j^T J \boldsymbol{\xi}_k \in \mathbb{Z}$

$$\boxed{A_{jk}} = \left(MJM^T\right)_{jk} \in \mathbb{Z} \quad \Rightarrow \quad \text{Symplectically integral lattices}$$

Symplectic Gram matrix

Each linearly independent generator “discretizes” 1 direction in phase space

$\Rightarrow$ Focus on full rank lattices: $\det M \neq 0 \Rightarrow d < \infty$ Logical dimension

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and logical operations

Logical Pauli operators :

all displacements that commute with stabilizers

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and logical operations

Logical Pauli operators :

all displacements that commute with stabilizers

$\updownarrow$

all vectors that have integer symplectic product with lattice vectors

$$\left\{ \boldsymbol{\xi}^{\perp} \in \mathbb{R}^{2n} \mid \left(\boldsymbol{\xi}^{\perp}\right)^{T} J \boldsymbol{\xi} \in \mathbb{Z} \, \forall \boldsymbol{\xi} \in \mathcal{L} \right\}$$

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and logical operations

Logical Pauli operators :

all displacements that commute with stabilizers

$\updownarrow$

all vectors that have integer symplectic product with lattice vectors

$\Rightarrow$ (symplectically) dual lattice $\mathcal{L}^{\perp} = \left\{ \boldsymbol{\xi}^{\perp} \in \mathbb{R}^{2n} \mid \left(\boldsymbol{\xi}^{\perp}\right)^{T} J \boldsymbol{\xi} \in \mathbb{Z}\, \forall \boldsymbol{\xi} \in \mathcal{L} \right\}$

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and logical operations

Logical Pauli operators :

all displacements that commute with stabilizers

$\updownarrow$

all vectors that have integer symplectic product with lattice vectors

$\Rightarrow$ (symplectically) dual lattice $\mathcal{L}^\perp = \left\{ \boldsymbol{\xi}^\perp \in \mathbb{R}^{2n} \mid \left(\boldsymbol{\xi}^\perp\right)^T J \boldsymbol{\xi} \in \mathbb{Z} \, \forall \boldsymbol{\xi} \in \mathcal{L} \right\}$

Dual lattice generator (*finite logical dimension*) : $M^\perp = (JM^T)^{-1}$

Gottesman, Kitaev, Preskill PRA 64 (2001)

Lattices and logical operations

Logical Pauli operators :

all displacements that commute with stabilizers

$\updownarrow$

all vectors that have integer symplectic product with lattice vectors

$\Rightarrow$ (symplectically) dual lattice $\mathcal{L}^{\perp} = \left\{ \boldsymbol{\xi}^{\perp} \in \mathbb{R}^{2n} \mid \left(\boldsymbol{\xi}^{\perp}\right)^{T} J \boldsymbol{\xi} \in \mathbb{Z}\, \forall \boldsymbol{\xi} \in \mathcal{L} \right\}$

Dual lattice generator (*finite logical dimension*) : $M^{\perp} = (JM^{T})^{-1}$

One can show that the logical dimension *d* is computed from:

$$d^2 = |\mathcal{L}^{\perp}/\mathcal{L}| = |\det M|/|\det M^{\perp}| = \det A = (\det M)^2$$

Gottesman, Kitaev, Preskill PRA 64 (2001)

Part 3 : Results

Lattice bases

Basis (stabilizer generators) not unique given lattice (code)

$$\mathcal{S} = \langle D\left(\boldsymbol{\xi}_1\right), \ldots, D\left(\boldsymbol{\xi}_{2n}\right)\rangle$$

$$M = \left(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n}\right)^T$$

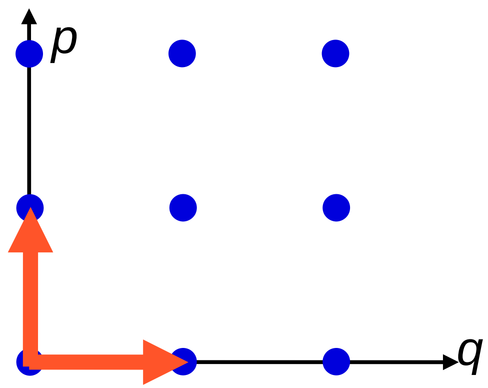
Lattice bases

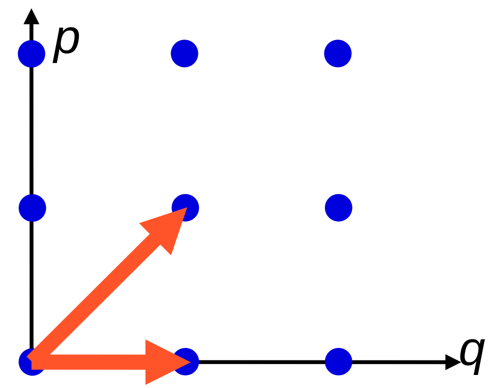
Basis (stabilizer generators) not unique given lattice (code)

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle$$

$$M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

<u>Example</u>: square lattice





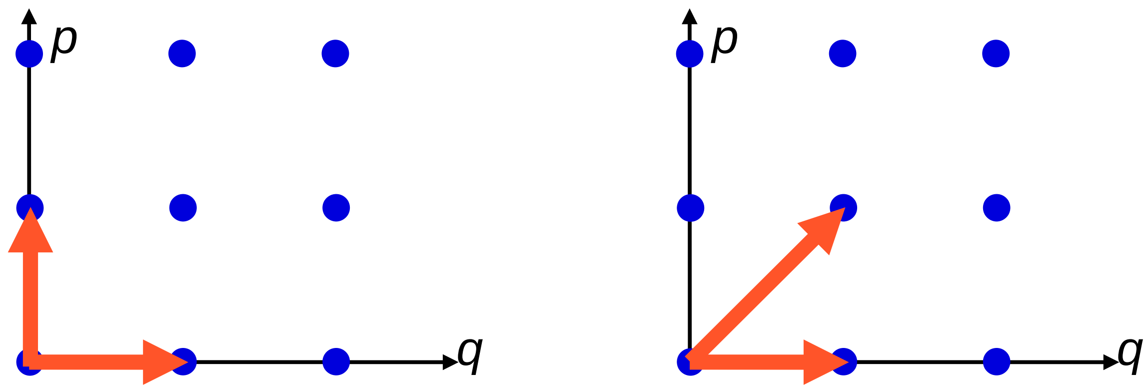
Lattice bases

Basis (stabilizer generators) not unique given lattice (code)

$$\mathcal{S} = \left\langle D\left(\boldsymbol{\xi}_1\right), \ldots, D\left(\boldsymbol{\xi}_{2n}\right)\right\rangle$$

$$M = \left(\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n}\right)^T$$

<u>Example</u>: square lattice



⇒ Exploit basis choice and properties to study codes

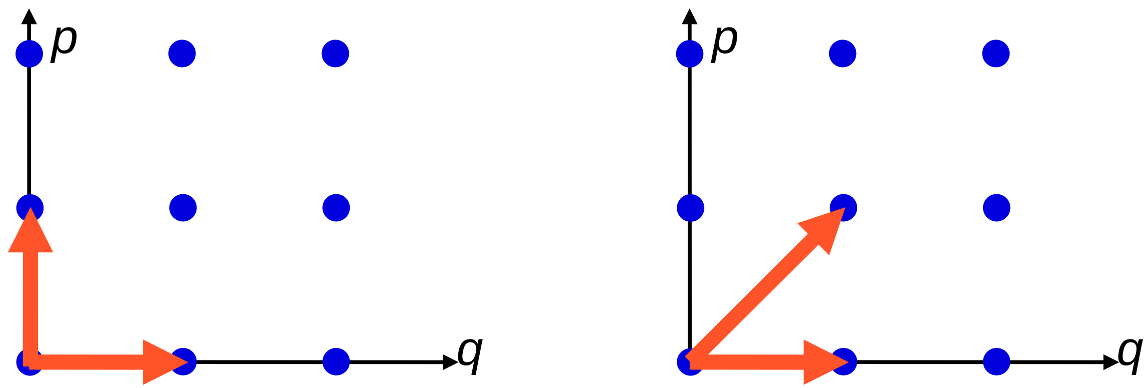
Lattice bases

Basis (stabilizer generators) not unique given lattice (code)

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle$$

$$M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

Example: square lattice



⇒ Exploit basis choice and properties to study codes

Theorem (Hadamard's bound):

$$\boxed{\text{Let } C = \max_j ||\boldsymbol{\xi}_j||,\ d = 2^k = \det M. \text{ Then } k \leq 2n \log_2 C}$$

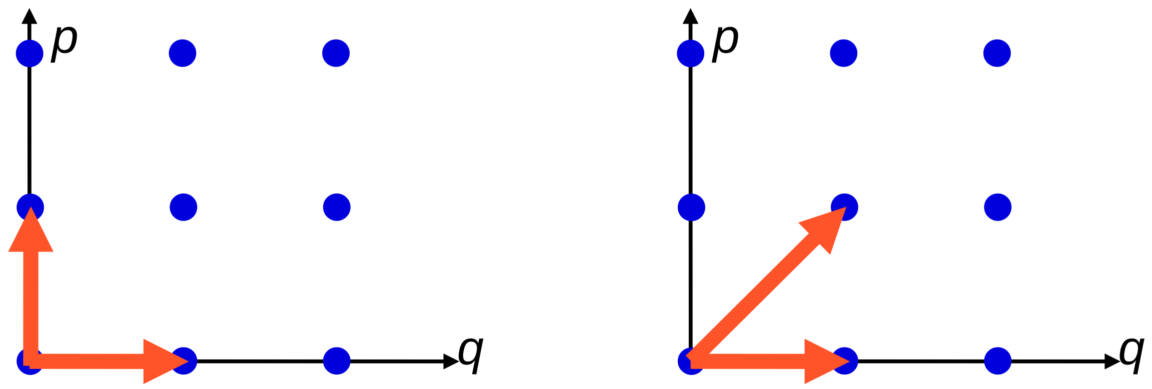
Lattice bases

Basis (stabilizer generators) not unique given lattice (code)

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle$$

$$M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

<u>Example</u>: square lattice



⇨ Exploit basis choice and properties to study codes

Theorem (Hadamard's bound):

$$\boxed{\text{Let } C = \max_j ||\boldsymbol{\xi}_j||,\ d = 2^k = \det M. \text{ Then } k \leq 2n \log_2 C}$$

Intuitively : $C \sim$ maximum interaction strength with ancilla to measure a stabilizer generator

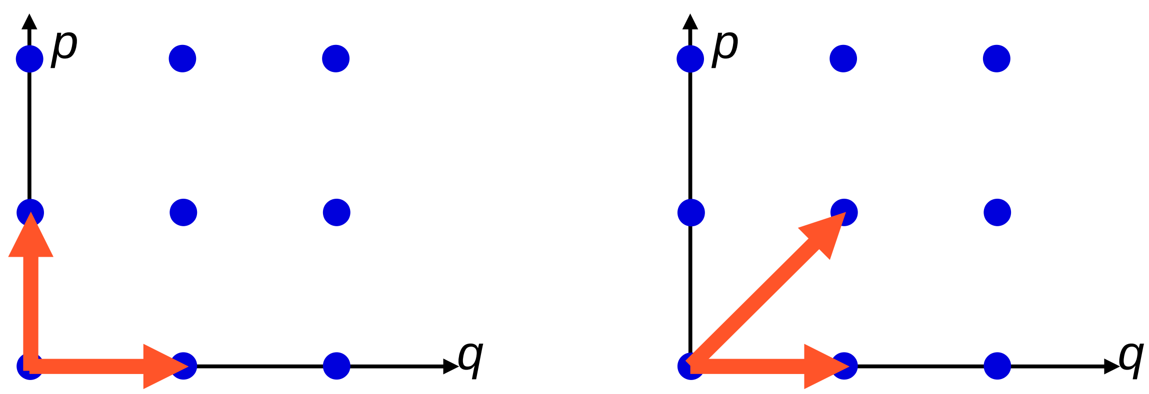
Lattice bases

Basis (stabilizer generators) not unique given lattice (code)

$$\mathcal{S} = \langle D(\boldsymbol{\xi}_1), \ldots, D(\boldsymbol{\xi}_{2n}) \rangle$$

$$M = (\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{2n})^T$$

<u>Example</u>: square lattice



⇨ Exploit basis choice and properties to study codes

Theorem (Hadamard's bound):

$$\boxed{\text{Let } C = \max_j ||\boldsymbol{\xi}_j||,\ d = 2^k = \det M. \text{ Then } k \leq 2n \log_2 C}$$

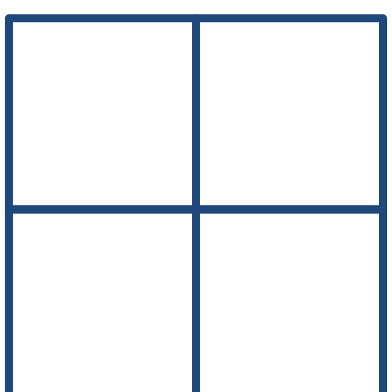
Intuitively : $C \sim$ maximum interaction strength with ancilla to measure a stabilizer generator

⇨ Encoding ratio related to "hardness" of measuring generators *in the chosen basis*

Minimal stabilizer sets for concatenated codes

Concatenation :

Example : L=3 GKP-surface code (*n* = *9* modes, logical dimension *d* = *2*)

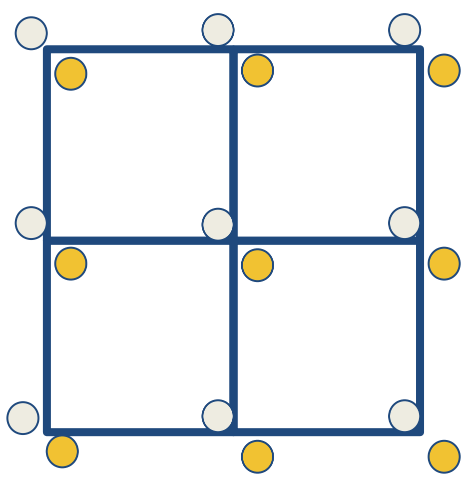


qubits/modes on vertices

Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension d = 2^n)
(think 1 square lattice for each mode) → $2n$ generators

Example : L=3 GKP-surface code (n = 9 modes, logical dimension d = 2)



qubits/modes on vertices

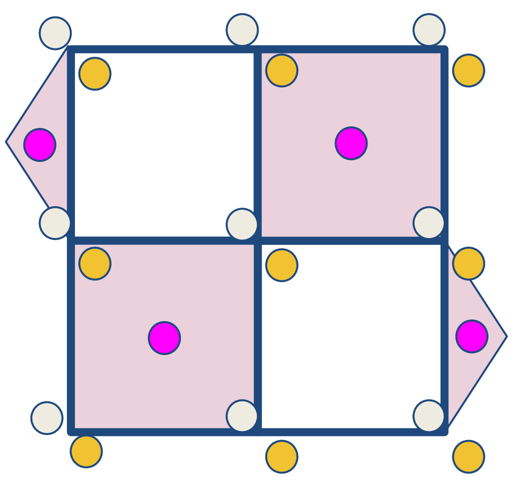
○ $\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$
● $\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$
GKP qubits

Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension d = 2^n)
(think 1 square lattice for each mode) → $2n$ generators

2) <u>enlarge stabilizer</u> to include additional <u>*logical*</u> Pauli strings
↔ displacements in <u>*dual lattice*</u>

Example : L=3 GKP-surface code (n = 9 modes, logical dimension d = 2)



qubits/modes on vertices

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$

$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$

GKP qubits

$$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$$

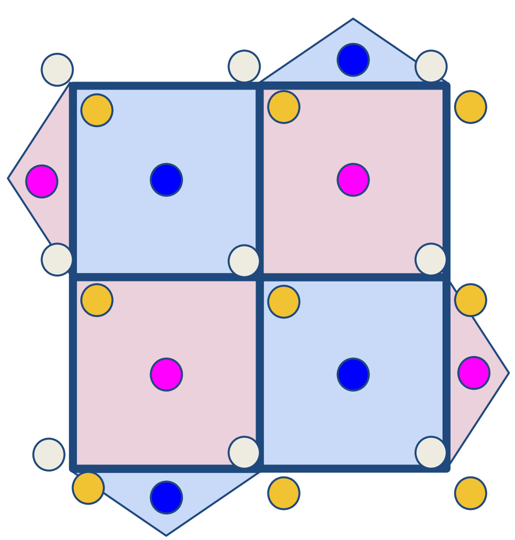
Higher level code

Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension $d = 2^n$)
(think 1 square lattice for each mode) $\rightarrow$ $2n$ generators

2) enlarge stabilizer to include additional *logical* Pauli strings
$\leftrightarrow$ displacements in *dual lattice*

Example : L=3 GKP-surface code (n = 9 modes, logical dimension d = 2)



qubits/modes on vertices

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$

$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$

GKP qubits

$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$

$\prod_{l(j)} Z^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\eta}_{l(j)}\right)$

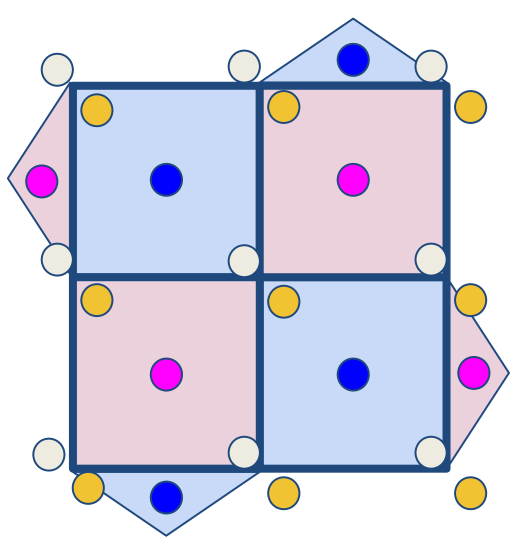
Higher level code

Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension $d = 2^n$)
(think 1 square lattice for each mode) → $2n$ generators

2) <u>enlarge stabilizer</u> to include additional <u>*logical*</u> Pauli strings
↔ displacements in <u>*dual lattice*</u>

Example : L=3 GKP-surface code (n = 9 modes, logical dimension d = 2)



qubits/modes on vertices

$2n + n - 1 = \mathbf{26}$ stab.

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$

$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$

GKP qubits

$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$

$\prod_{l(j)} Z^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\eta}_{l(j)}\right)$

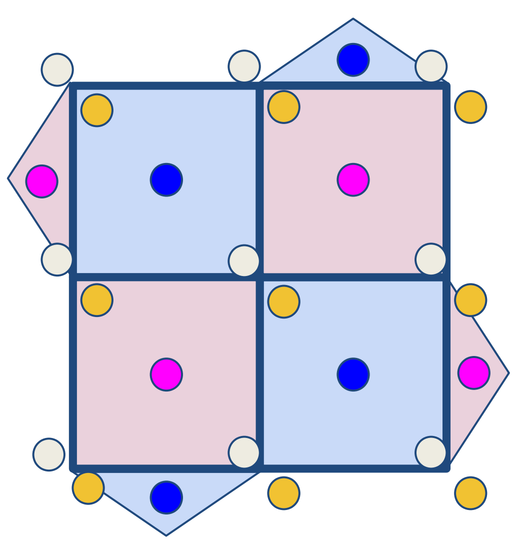
Higher level code

Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension $d = 2^n$)
(think 1 square lattice for each mode) → $2n$ generators
2) <u>enlarge stabilizer</u> to include additional <u>*logical*</u> Pauli strings
↔ displacements in <u>*dual lattice*</u>

Overall code still has to be stabilized by a **(new) lattice** of displacements (with smaller d)

Example : L=3 GKP-surface code ($n = 9$ modes, logical dimension $d = 2$)



qubits/modes on vertices

$2n + n - 1 =$ ***26*** stab.

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$
$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$ GKP qubits

$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$

$\prod_{l(j)} Z^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\eta}_{l(j)}\right)$

Higher level code

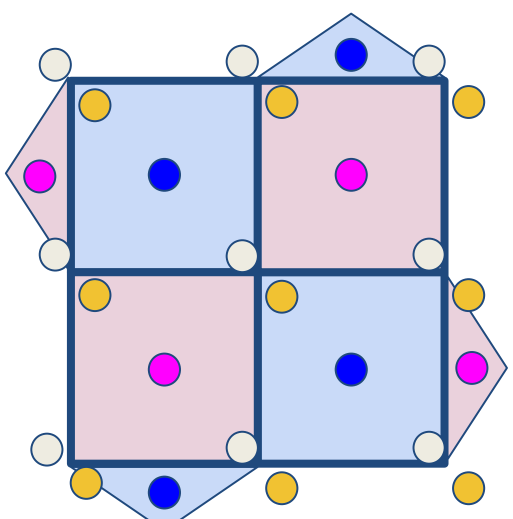
Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension $d = 2^n$)
(think 1 square lattice for each mode) $\rightarrow$ $2n$ generators

2) <u>enlarge stabilizer</u> to include additional <u>*logical*</u> Pauli strings
$\leftrightarrow$ displacements in <u>*dual lattice*</u>

Overall code still has to be stabilized by a **(new) lattice** of displacements (with smaller d)
But : $2n$ vectors are always sufficient to define a full dimensional lattice !

Example : L=3 GKP-surface code (n = 9 modes, logical dimension d = 2)



qubits/modes on vertices

2n + n - 1 = ***26*** stab.

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$

$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$

GKP qubits

$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$

$\prod_{l(j)} Z^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\eta}_{l(j)}\right)$

Higher level code

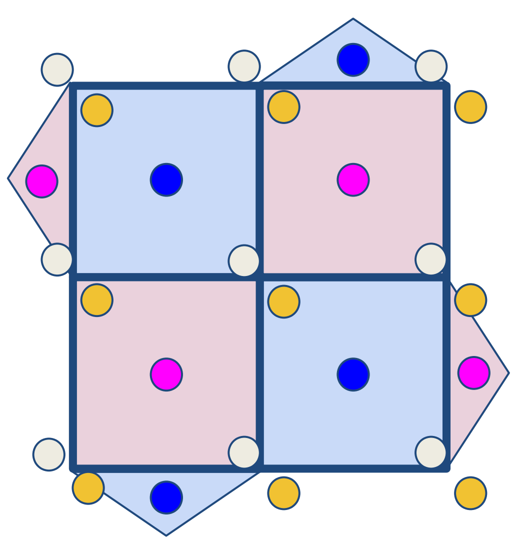
Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of n modes (logical dimension $d = 2^n$)
(think 1 square lattice for each mode) → $2n$ generators

2) <u>enlarge stabilizer</u> to include additional <u>*logical*</u> Pauli strings
↔ displacements in <u>*dual lattice*</u>

Overall code still has to be stabilized by a **(new) lattice** of displacements (with smaller d)
But : $2n$ vectors are always sufficient to define a full dimensional lattice !

Example : L=3 GKP-surface code ($n = 9$ modes, logical dimension $d = 2$)



qubits/modes on vertices

$2n + n - 1 =$ ***26*** stab.

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$

$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$

GKP qubits

$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$

$\prod_{l(j)} Z^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\eta}_{l(j)}\right)$

Higher level code

$$M_{\mathrm{conc}} = \begin{pmatrix} M_{\mathrm{GKP}} \\ M_{\mathrm{Q}} \end{pmatrix}$$

⇩ lattice basis reduction

M_{min}

$2n =$ ***18*** stab. gen.

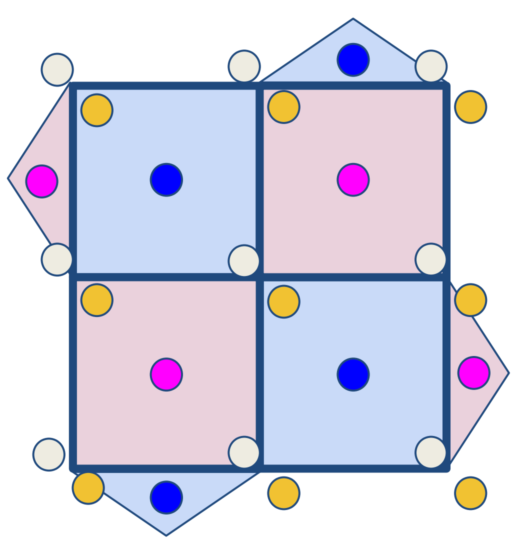
can do respecting weights
and geometric locality!

Minimal stabilizer sets for concatenated codes

Concatenation : 1) fix a qubit in each of *n* modes (logical dimension $d = 2^n$)
(think 1 square lattice for each mode) → $2n$ generators
2) <u>enlarge stabilizer</u> to include additional <u>*logical*</u> Pauli strings
↔ displacements in <u>*dual lattice*</u>

Overall code still has to be stabilized by a **(new) lattice** of displacements (with smaller *d*)
But : $2n$ vectors are always sufficient to define a full dimensional lattice !

Example : L=3 GKP-surface code ($n = 9$ modes, logical dimension $d = 2$)



qubits/modes on vertices

$2n + n - 1 =$ ***26*** stab.

$\exp\left(i2\sqrt{\pi}\hat{q}_j\right)$

$\exp\left(-i2\sqrt{\pi}\hat{p}_j\right)$

GKP qubits

$\prod_{l(j)} X^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\xi}_{l(j)}\right)$

$\prod_{l(j)} Z^{(L)}_{l(j)} = \prod_{l(j)} D\left(\boldsymbol{\eta}_{l(j)}\right)$

Higher level code

$$M_{\mathrm{conc}} = \begin{pmatrix} M_{\mathrm{GKP}} \\ M_{\mathrm{Q}} \end{pmatrix}$$

lattice basis reduction

$M_{\min}$

$2n =$ ***18*** stab. gen.

can do respecting weights
and geometric locality!

less measurements
per EC cycle

Symplectic operations

$$U_S = \exp\left(-i\hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}\right)$$

$$U_S \hat{\boldsymbol{x}} U_S^\dagger = S\hat{\boldsymbol{x}} \qquad S \in \mathrm{Sp}(2n), \quad SJS^T = J$$

Symplectic operations

$$U_S = \exp\left(-i\hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}\right)$$

$$U_S \hat{\boldsymbol{x}} U_S^\dagger = S\hat{\boldsymbol{x}} \qquad S \in \mathrm{Sp}(2n), \quad SJS^T = J$$

$$A = MJM^T$$

Theorem (symplectically equivalent codes):

$$\boxed{\text{Given } \mathcal{L}(M), \mathcal{L}(N),\ \exists S \mid M = NS^T \text{ iff } MJM^T = NJN^T \text{ (in canonical form)}}$$

Multi-mode generalization of *Hänggli, Heinze, König, PRA 102 (2020)*

$$A_{M,N} = \begin{pmatrix} 0 & D \\ -D & 0 \end{pmatrix}$$

Symplectic operations

$$U_S = \exp\left(-i\hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}\right)$$

$$U_S \hat{\boldsymbol{x}} U_S^\dagger = S\hat{\boldsymbol{x}} \qquad S \in \mathrm{Sp}(2n), \quad SJS^T = J$$

$$A = MJM^T$$

Theorem (symplectically equivalent codes):

$$\boxed{\text{Given } \mathcal{L}(M), \mathcal{L}(N),\ \exists S \mid M = NS^T \text{ iff } MJM^T = NJN^T \text{ (in canonical form)}}$$

Multi-mode generalization of *Hänggli, Heinze, König, PRA 102 (2020)*

$$A_{M,N} = \begin{pmatrix} 0 & D \\ -D & 0 \end{pmatrix}$$

Intuition : logical dimensions are distributed in the same way across oscillators

Symplectic operations

$$U_S = \exp\left(-i\hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}\right)$$

$$U_S \hat{\boldsymbol{x}} U_S^\dagger = S\hat{\boldsymbol{x}} \qquad S \in \mathrm{Sp}(2n), \quad SJS^T = J$$

$$A = MJM^T$$

Theorem (symplectically equivalent codes):

$$\boxed{\text{Given } \mathcal{L}(M), \mathcal{L}(N), \ \exists S \mid M = NS^T \text{ iff } MJM^T = NJN^T \text{ (in canonical form)}}$$

Multi-mode generalization of *Hänggli, Heinze, König, PRA 102 (2020)*

$$A_{M,N} = \begin{pmatrix} 0 & D \\ -D & 0 \end{pmatrix}$$

Intuition : logical dimensions are distributed in the same way across oscillators

Corollary :

$$\boxed{\text{Any code with } d = 2 \text{ is s.e. to } \mathcal{S}_\square^{(2)} = \left\langle e^{i2\sqrt{\pi}\hat{q}_1}, e^{-i2\sqrt{\pi}\hat{p}_1}, e^{i\sqrt{\pi}\hat{q}_j}, e^{i\sqrt{\pi}\hat{p}_j} \right\rangle, \ j > 1}$$

one qubit in mode 1, a fixed state on others

Symplectic operations

$$U_S = \exp\left(-i\hat{\boldsymbol{x}}^T H \hat{\boldsymbol{x}}\right)$$

$$U_S \hat{\boldsymbol{x}} U_S^\dagger = S\hat{\boldsymbol{x}} \qquad S \in \mathrm{Sp}(2n), \quad SJS^T = J$$

$$A = MJM^T$$

Theorem (symplectically equivalent codes):

$$\boxed{\text{Given } \mathcal{L}(M), \mathcal{L}(N),\ \exists S \mid M = NS^T \text{ iff } MJM^T = NJN^T \text{ (in canonical form)}}$$

Multi-mode generalization of *Hänggli, Heinze, König, PRA 102 (2020)*

$$A_{M,N} = \begin{pmatrix} 0 & D \\ -D & 0 \end{pmatrix}$$

Intuition : logical dimensions are distributed in the same way across oscillators

Corollary :

$$\boxed{\text{Any code with } d = 2 \text{ is s.e. to } \mathcal{S}_{\square}^{(2)} = \left\langle e^{i2\sqrt{\pi}\hat{q}_1}, e^{-i2\sqrt{\pi}\hat{p}_1}, e^{i\sqrt{\pi}\hat{q}_j}, e^{i\sqrt{\pi}\hat{p}_j} \right\rangle,\ j > 1}$$

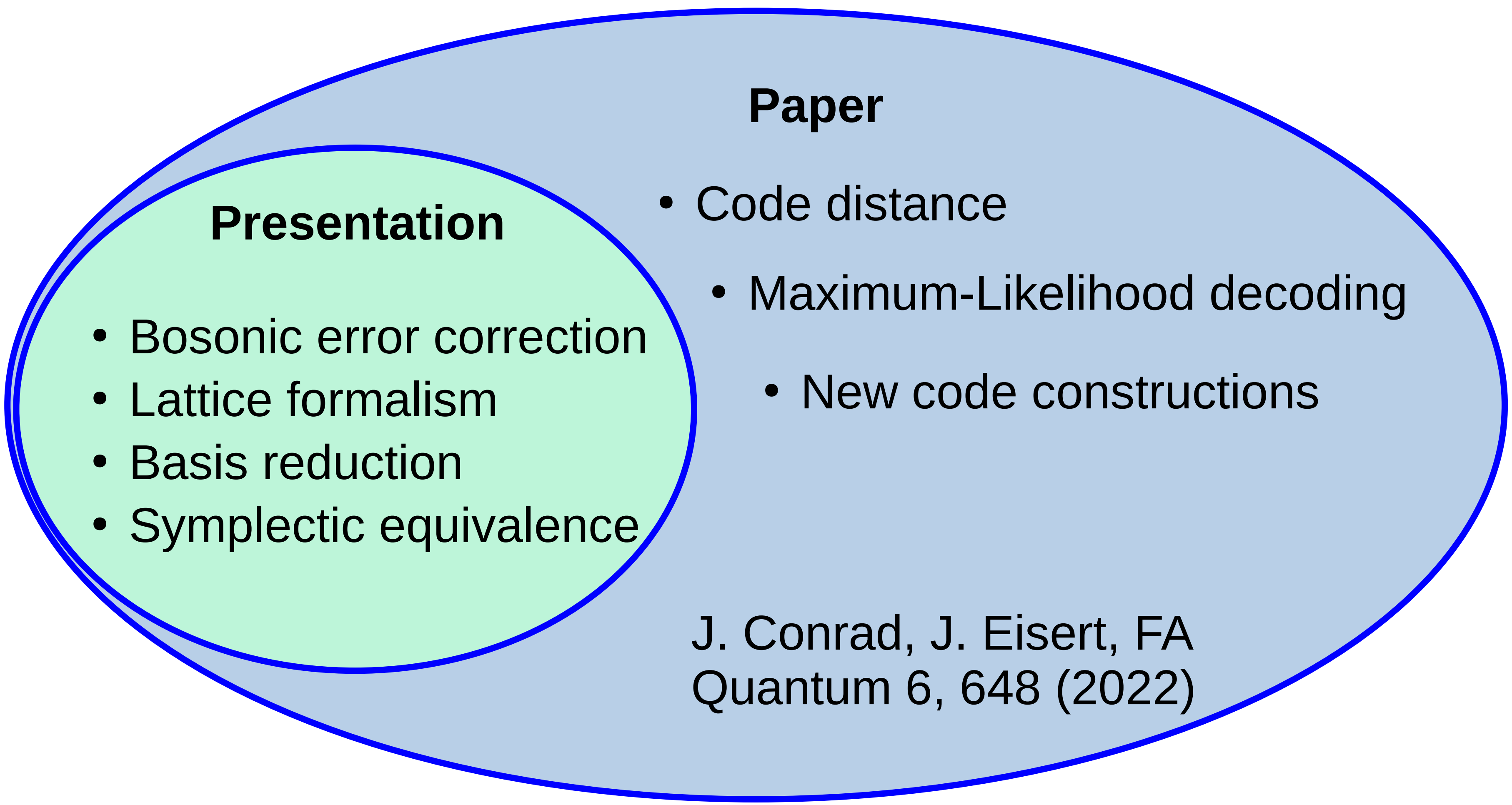
one qubit in mode 1, a fixed state on others

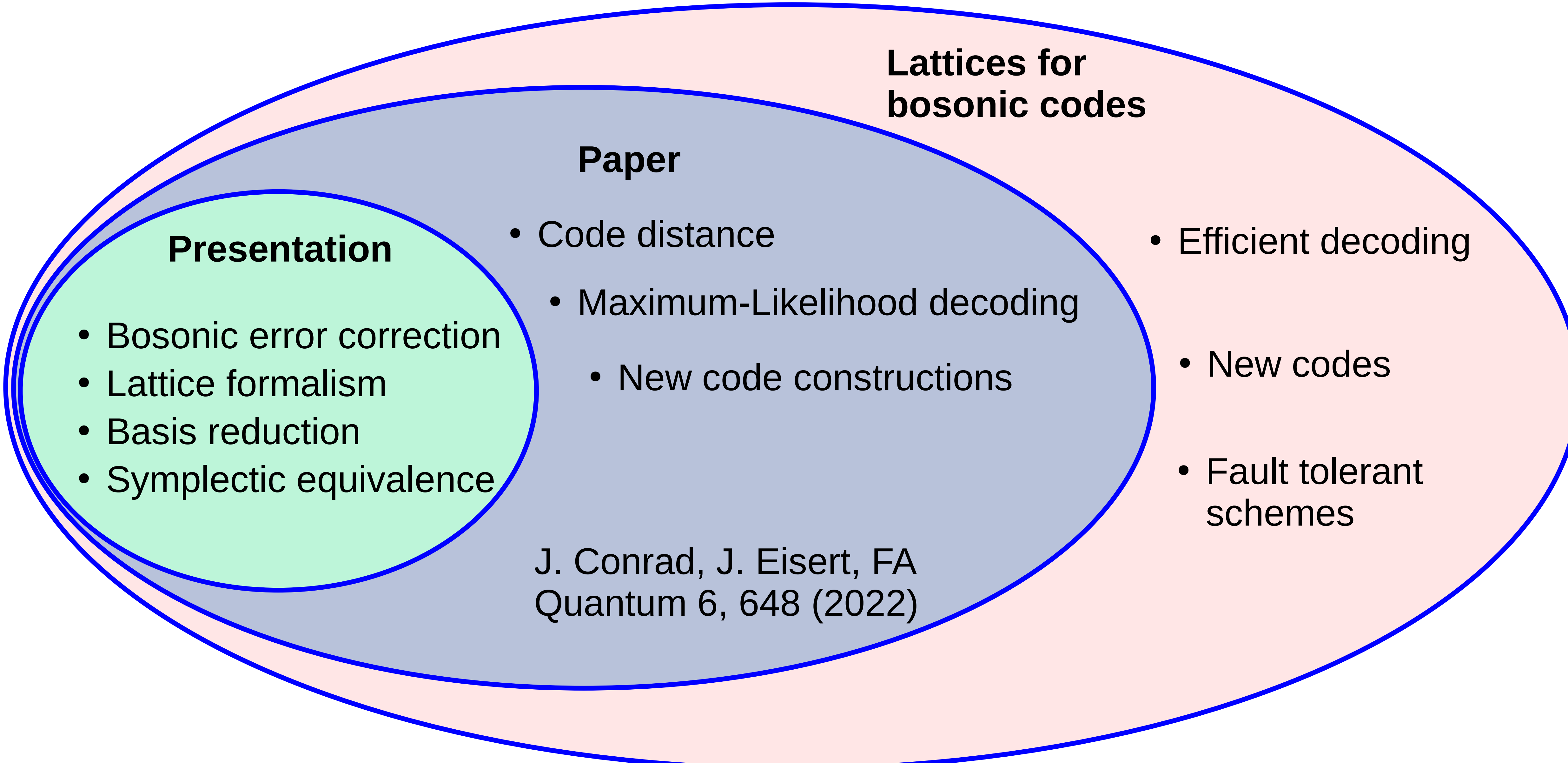
- **Generalizes** to higher logical dimensions
- Can be used to **bound distance** of (any) given grid code (see paper)

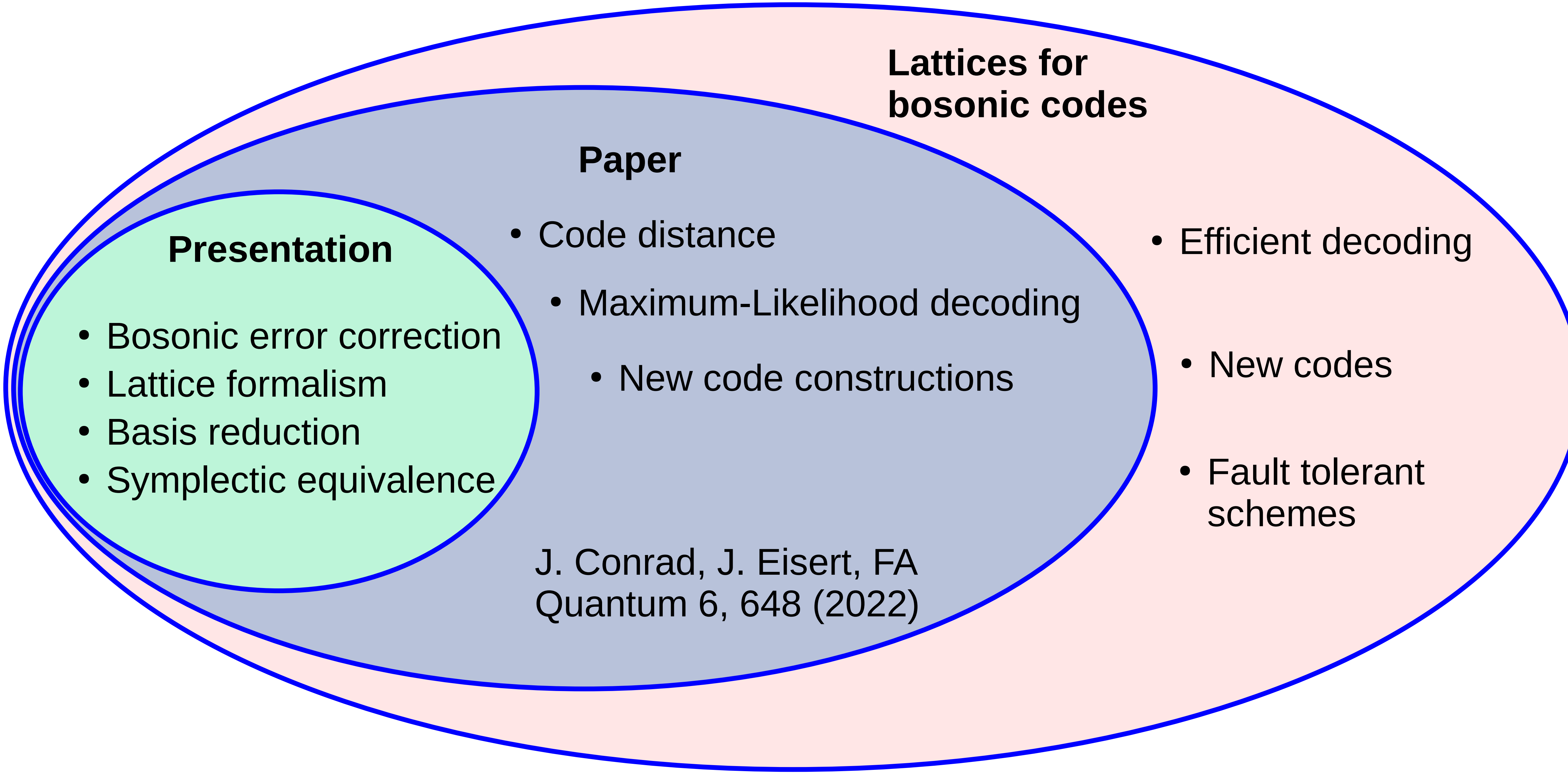
Part 4 : Conclusions

Presentation

- Bosonic error correction
- Lattice formalism
- Basis reduction
- Symplectic equivalence


Paper
Presentation
• Code distance
• Maximum-Likelihood decoding
• New code constructions
• Bosonic error correction
• Lattice formalism
• Basis reduction
• Symplectic equivalence
J. Conrad, J. Eisert, FA
Quantum 6, 648 (2022)


Lattices for
bosonic codes
Paper
• Code distance
• Maximum-Likelihood decoding
• New code constructions
J. Conrad, J. Eisert, FA
Quantum 6, 648 (2022)
Presentation
• Bosonic error correction
• Lattice formalism
• Basis reduction
• Symplectic equivalence
• Efficient decoding
• New codes
• Fault tolerant
schemes


Lattices for bosonic codes
Paper
Presentation
• Bosonic error correction
• Lattice formalism
• Basis reduction
• Symplectic equivalence
• Code distance
• Maximum-Likelihood decoding
• New code constructions
J. Conrad, J. Eisert, FA
Quantum 6, 648 (2022)
• Efficient decoding
• New codes
• Fault tolerant schemes

Thank you!