Bosonic Quantum Error Correcting Codes

Ronak Ramachandran

May 14, 2022

Abstract

In this survey paper, we take a closer look at Bosonic quantum error correcting codes. First, we give an overview of quantum error correction, then we use ideas from this overview to compare and contrast a selection of Bosonic codes—most prominently the GKP code. Bosonic codes address new kinds of errors, different from the errors that stabilizer codes handle. For instance, Fock-state Bosonic codes handle photon gain, loss, and dephasing errors, while GKP codes handle displacements in phase space. When possible, we try to explain notions of rate and distance, but these can be hard to compare for codes over different Hilbert spaces and error models.

1 Introduction

Error correction is the art of representing information in forms that are robust to a specific class of errors. After such an error occurs, we'd like to have any easy way to detect and/or correct the error.

In order to achieve this, we must represent information redundantly. For instance, the $[7, 4]$ Hamming code encodes 4 classical bits of information in 7 classical bits, while Shor's 9-qubit code encodes 1 qubit in 9 qubits.

We call the information we'd like to encode our 'message' or 'logical state', and we call encoded states with no error 'codewords' or 'codestates'. The set of all codewords is called the codespace. In the $[7, 4]$ Hamming code, for instance, a message would be 4 bits long, while a codeword would be 7 bits long, and the codespace would contain 16 codewords corresponding with the $2^4 = 16$ possible 4-bit messages.

Due to redundancy, the codespace is embedded in a much larger space of possible states. States outside the codespace are called error states. The goal of error correction is to design a codespace so that errors within our error model take codestates to error states from which the original codestates are recoverable.

In the case of quantum error correcting codes, we'd also like to be able to perform gates on the encoded states. This is necessary for the quantum threshold theorem to hold, which states that it's possible to do encoded quantum computation with arbitrarily low error so long as the error per operation is below a certain threshold [KLZ98]. This theorem is what gives us hope that our current noisy hardware has a chance of leading to fault-tolerant quantum computation (FTQC).

In general, specifying a Quantum Error Correcting Code (QECC) requires 4 steps: (1) Define an error model, (2) Specify a codespace through stabilizing operations (syndrome measurements), (3) Show how to perform a universal set of gates on encoded states, and (4) Show how to prepare and decode codestates.

1.1 Error Models

Defining an error model means specifying the set $\mathcal{E}$ of errors which we aim to correct. Note that errors do not have to be unitary. Errors need to satisfy two requirements:

$$\langle\psi_i| E_a^\dagger E_b |\psi_j\rangle = 0 \tag{1}$$

$$\langle\psi_i| E_a^\dagger E_b |\psi_i\rangle = \langle\psi_j| E_a^\dagger E_b |\psi_j\rangle \tag{2}$$

for all $E_a, E_b \in \mathcal{E}$ and $|\psi_i\rangle, |\psi_j\rangle \in C$ [Got97], where $i \neq j$ and C is the codespace. Intuitively, the first requirement means that no two errors can make one codestate look like another, while the second requirement means that the operations required to correct errors are independent of what logical state the system is in.

Note the following crucial theorem:

Theorem 1. *If a quantum code corrects errors A and B, it also corrects any linear combination of A and B. In particular, if it corrects all weight t Pauli errors, then the code corrects all t-qubit errors. [Got10]*

For this reason, the codes that follow often focus on correcting a set of errors $\mathcal{E}$ that form an operator basis for operators (not necessarily Pauli errors) of a certain weight (not necessarily measured in qubits).

1.2 Syndrome measurements

An operation S is said to stabilize a state $|c\rangle$ iff $S|c\rangle = |c\rangle$ (here, global phase matters). Specifying a set $\mathcal{S}$ of stabilizing operations defines a codespace

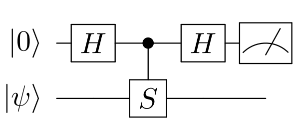
$$C = \{|c\rangle : \forall S \in \mathcal{S}\ S|c\rangle = |c\rangle\}.$$

The codespace is the subspace of the Hilbert space in which all codestates (codewords) live. For instance, in the 5-qubit stabilizer code, $\mathcal{S}$ is the group generated by the 5-qubit Pauli tensors $XZZXI$, $IXZZX$, $XIXZZ$, and $ZXIXZ$, and C is the set of all states stabilized by all elements of that group.

Note that all quantum error correcting codes have stabilizers, not just (the somewhat confusingly named) stabilizer codes. Stabilizer codes are stabilized by Pauli operations, while non-stabilizer codes are stabilized by non-Pauli operations.

Codes often have two dimensional codespaces—this corresponds with leaving enough degrees of freedom unstabilized in order to embed a single logical qubit in the codespace. The goal is to pick a codespace such that errors in $\mathcal{E}$ look very different from gates applied to the logical states encoded in the codespace.

To detect and correct errors, we can perform syndrome measurements of the form:



for each $S \in \mathcal{S}$. This is similar to the swap test. Note that for stabilizer codes, $S\left|c^\perp\right\rangle = -\left|c^\perp\right\rangle$ for all $\left|c^\perp\right\rangle \perp C$ and $S \in \mathcal{S}$, so the probability of measuring 0 for a state $|\psi\rangle = \alpha|c\rangle + \beta\left|c^\perp\right\rangle$ where $|c\rangle \in C$ and $\left|c^\perp\right\rangle \perp C$ is $|\alpha|^2$. This nice property does not necessarily hold for other sets of stabilizers, for instance, those used in the GKP code.

An alternative but equivalent way to define syndrome measurements is to measure $\langle\psi| E_a^\dagger E_b |\psi\rangle$ for all $E_a, E_b \in \mathcal{E}$ [Got97].

1.3 Logical Gates

Applying logical gates to the quantum state embedded in the codespace C can be very code-dependent. We will see some examples in the sections that follow.

1.4 Encoding and Decoding

Preparing the logical states of a code and decoding information at the end of an encoded computation can both also be very code-dependent. We will not spend much time discussing either in this paper, but both are crucial for making a code useful. For instance, preparing GKP codestates is one of the central difficulties behind creating a scalable photonic fault-tolerant quantum computer [Bou et. al. 21].

2 Fock State Bosonic Codes

One basis for Bosonic states is the Fock (number) basis. Fock basis states are differentiated by the number of excitations in each mode. These excitation numbers can refer to the discrete energy levels of a list of oscillators or the count of photons in a list of modes, but the math for both is the same. For most of this paper, we will treat excitation numbers as photon counts.

Fock states have three new kinds of errors: photon gain, photon loss, and dephasing errors. Photon gains and losses can be caused by leakage to and from the environment or from spontaneous particle-anti-particle creation or annihilation. Dephasing errors are a form of decoherence resulting from noise in the phases of various Fock basis states. These errors are often expressed in terms of creation and annihilation operators, so a code which can correct L losses, G gains, and D dephasing errors would be able to correct the set of errors $\mathcal{E} = \{\mathbb{I}, \hat{a}, \ldots, \hat{a}^L, \hat{a}^\dagger, \ldots, (\hat{a}^\dagger)^G, \hat{a}\hat{a}^\dagger, \ldots, (\hat{a}\hat{a}^\dagger)^D\}$ [TCV20]. For brevity, $\hat{a}\hat{a}^\dagger$ is often denoted as $\hat{n}$.

2.1 Chuang-Leung-Yamamoto codes

Chuang-Leung-Yamamoto codes [CLY97] are a class of codes which embed k qubits into m oscillators with total excitation number (total number of particles) N. Codewords are either balanced (particles are evenly distributed in the modes) or are superpositions of unbalanced Fock states. These codes have a reasonable notion of rate and distance, and are often denoted as $[[N, m, 2^k, d]]$ codes. The distance d is defined as the minimum spacing between the Fock states that make up the codewords, where the spacing between states $u = |u_1, \ldots, u_m\rangle$ and $v = |v_1, \ldots, v_m\rangle$ is

$$\text{Spacing}(u, v) = \frac{1}{2}\sum_{i=1}^{m} |u_i - v_i|.$$

The rate of the code is defined to be

$$R = \frac{k}{m \log_2(N+1)}.$$

2.2 Wasilewski Banaszek Code

The Wasilewski Banaszek Code [WB07] is a simple example of a Chuang-Leung-Yamamoto code. Its codewords are

$$|L\rangle = \frac{1}{\sqrt{3}}(|300\rangle + |030\rangle + |003\rangle)$$
$$|H\rangle = |111\rangle .$$

This code protects against a single photon loss in any of its three modes. It is a $[[3, 3, 2, 2]]$ Chuang-Leung-Yamamoto code.

The stabilizers of the codespace are generated by

$$\Gamma_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & \omega & \omega^2 \\ 1 & \omega^2 & \omega \end{pmatrix}, \quad \Gamma_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \omega \end{pmatrix},$$

where $\omega = \exp(2\pi i/3)$. Both are easily implemented in beamsplitter networks. $\Gamma_2 = \text{QFT}_3$ is called a "tritter," and Γ_3 is a phaseshifter applied to one mode.

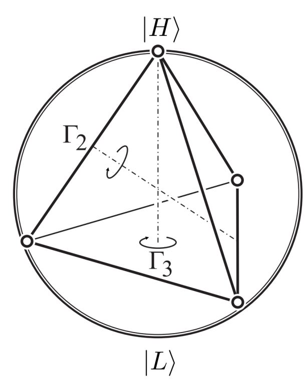


Figure 1: The stabilizers and the codespace can be visualized as the tetrahedral group. [WB07]

The original paper also demonstrates how to encode a qubit from the dual-rail representation into the Wasilewski Banaszek Code, and it shows how to construct beamsplitters that act on the encoded qubit, but we've omitted those details here for brevity.

2.3 Binomial Code

The Binomial code [Mich16] has codestates given by

$$|\bar{0}\rangle = \frac{1}{\sqrt{2^n}} \sum_{p,even}^{[0,N+1]} \sqrt{\binom{N+1}{p}} |p(S+1)\rangle$$
$$|\bar{1}\rangle = \frac{1}{\sqrt{2^n}} \sum_{p,odd}^{[0,N+1]} \sqrt{\binom{N+1}{p}} |p(S+1)\rangle .$$

If $S = L + G$, and $N = \max(L, G, 2D)$, and the max Fock number is $(N+1) \times (S+1)$, then this code can correct L photon losses, G photon gains, and D dephasing errors.

The lowest order binomial code is

$$|\bar{0}\rangle = \frac{|0\rangle + |4\rangle}{\sqrt{2}}$$
$$|\bar{1}\rangle = |2\rangle ,$$

and this protects against one photon loss, that is, the error set $\mathcal{E} = \{\mathbb{I}, \hat{a}\}$.

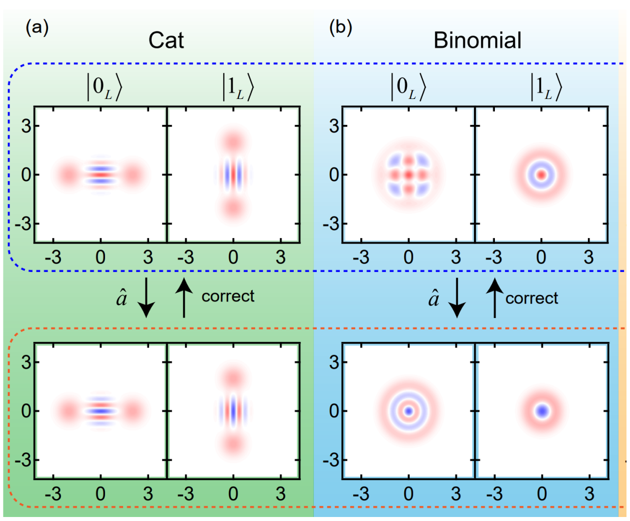


Figure 2: Wigner functions for Cat and Binomial codes.[CYWZS21]. For more on Wigner functions, see Appendix B

2.4 Cat Codes

After some thought, I see no way to improve over the explanations given in Wikipedia: cat state wiki.

3 CV Bosonic Codes

Continuous Variable (CV) quantum computing presents an entirely new set of errors: ε-sized (complex valued) displacements in phase space. In order to understand these errors, we'll need to define CV quantum computation.

3.1 From Qubits to Qudits to CV

Qubits are two-level systems with basis states $|0\rangle$ and $|1\rangle$. Qudits are d-level systems with basis states $|0\rangle$, $|1\rangle$, ..., $|d-1\rangle$. We can generalize the Pauli X and Z operators to X_d and Z_d, known as the "shift" and "clock" operators. These are defined so

$$X_d |n\rangle \mapsto |n+1\rangle$$
$$Z_d |n\rangle \mapsto \omega^n |n\rangle,$$

where addition is done in $\mathbb{F}_d$ and $\omega = \exp(2\pi i/d)$.

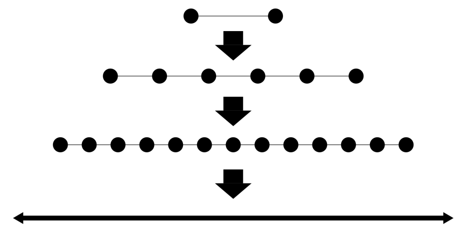
We can also look at the Fourier conjugate basis

$$|\omega^n\rangle := \text{QFT}_d |n\rangle = \frac{1}{\sqrt{d}} \sum_{m=0}^{d-1} \omega^{n \cdot m} |m\rangle,$$

where in the case of qubits, the Fourier conjugate basis consists of $|+\rangle$ and $|-\rangle$. Physicists often refer to the primal basis $|0\rangle, |1\rangle, \dots, |d-1\rangle$ as the "position quadrature," and the Fourier conjugate basis $|\omega^0\rangle, |\omega^1\rangle, |\omega^2\rangle, \dots, |\omega^{d-1}\rangle$ as the "momentum quadrature." Position is associated with the letter q, while momentum is associated with p.

One helpful tool for physicists is the Wigner function of a state, a quasiprobability distribution over phase space which simultaneously gives insight into the shape of the probability distributions for measuring a state in the position and momentum quadratures. Phase space (no relation to amplitude phases) is the abstract space where each point is specified by a position and momentum coordinate pair. For the rest of this paper, we will assume familiarity with the Wigner function. For more on the Wigner function, see Appendix B.

Continuous Variable quantum computing arises in the limit as d approaches infinity.



What this looks like is states defined by amplitudes at each position along the real number line. A number of interesting things happen in the limit. Knowing a state with certainty in the position basis would mean that state is an equal superposition of all infinitely many momentum basis states, which is not normalizable. Because of this, the best we can do is attempt to minimize uncertainty in both bases, resulting in Gaussian states. Reducing a Gaussian's uncertainty in one quadrature (squeezing) increases its uncertainty in the other quadrature.

Next, note that the shift and clock operators in the limit commute, because $Z_d X_d = \omega X_d Z_d$, and $\omega = \exp(2\pi i/d) \to 1$. This allows us to define a new continuous operation generalizing both, the displacement operator:

$$D(\alpha) := \exp(\alpha \hat{a}^\dagger - \alpha^* \hat{a}).$$

X_d^k shifts a qudit's discrete Wigner function k units in the position basis, and Z_d^k shifts it k units in the momentum basis. Similarly, the displacement operator shifts a state's Wigner function by $\text{Re}(\alpha)$ in the position basis and $\text{Im}(\alpha)$ in the momentum basis. Displacing the vacuum state by α yields the coherent state

$$|\alpha\rangle = \frac{1}{\sqrt{e^{|\alpha|^2}}} \sum_{n=0}^{\infty} \frac{\alpha^n}{\sqrt{n!}} |n\rangle,$$

where $|n\rangle$ is expressed in the Fock basis.

Note that even arbitrarily small magnitude displacements result in an orthogonal state. This is very different from discrete variable quantum computing, where small rotations result in states very close to the initial state.

One important subclass of CV are the Gaussian states (all states with non-negative Wigner functions) and Gaussian operations (displacement, squeezing, and linear optics). These are classically simulable. To state this a different way, any CV computation with a non-negative Wigner function that stays non-negative throughout is classically simulable. [ME12]

The convenience of stabilizer codes comes from the fact that $\mathbb{F}_2$ is cyclic with order 2, so stabilizing operations are easy to find. In CV, we work over $\mathbb{R}$, so we need to get creative with our states in order to define stabilizing operations (and consequently the codespaces and syndrome measurements necessary for a QECC). The GKP code achieves this.

3.2 The GKP Code

GKP codes were invented in 2001 by Gottesman, Kitaev, and Preskill [GKP01]. The ideal GKP codes are Dirac combs with codestates (expressed in position space) defined by

$$|\mu\rangle_{\text{gkp}} = \sum_n \left|(2n+\mu)\sqrt{\pi}\right\rangle_q, \;\; \mu \in \{0,1\}.$$

Here, the states $|(2n+\mu)\sqrt{\pi}\rangle_q$ are idealized Dirac delta functions, which are 0 everywhere except at

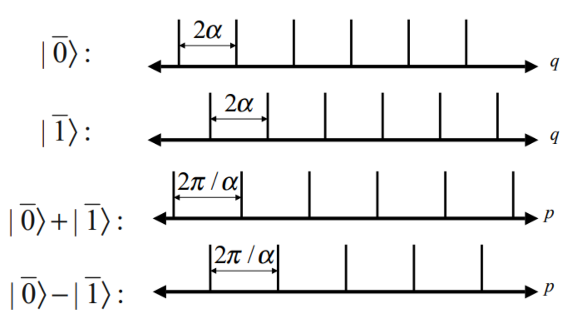


Figure 3: These states are called Dirac combs because when graphed in the q (position) or p (momentum) bases, the regularly spaced Dirac delta functions look like combs. For unbiased error correction, we choose $\alpha = \sqrt{\pi}$.

Stabilizing operations for the ideal GKP code are

$$\hat{S}_q = D(i\sqrt{2\pi}) = e^{i2\sqrt{\pi}\hat{q}}$$
$$\hat{S}_p = D(\sqrt{2\pi}) = e^{-i2\sqrt{\pi}\hat{p}}.$$

When we make the associated syndrome measurements, we ideally snap back to the nearest codeword. This means that the ideal GKP code can perfectly correct from displacement errors of magnitude less than $\sqrt{\pi/2}$. For larger displacements, syndrome measurements would still snap the state to the nearest codeword, but the nearest codeword might not be the correct logical state. In order to correct these logical errors, we would need to concatenate the GKP code with a conventional qubit error correcting code [Wang17].

There is one other complication with syndrome measurements: in CV, even the smallest displacement changes the state to something orthogonal. This means that states orthogonal to the codespace are no longer all in the -1 eigenspace of the stabilizers, so syndrome measurements don't behave quite as nicely as they did for stabilizer codes. In practice syndrome measurements are often performed using something called homodyne measurements [Bou et. al. 21].

Unfortunately, the ideal GKP code is not normalizable, and therefore can't exist. In order to realize the code, we need to relax the ideal code in two ways: (1) we need to loosen each tooth of the Dirac comb from a Dirac delta function to a (hopefully very thin) Gaussian, and (2) we need to wrap the whole wave function in a (hopefully very broad) Gaussian envelope with peaks of the comb fading the further we get from $(q,p) = (0,0)$. After these corrections, the Wigner function of the GKP code looks like this:

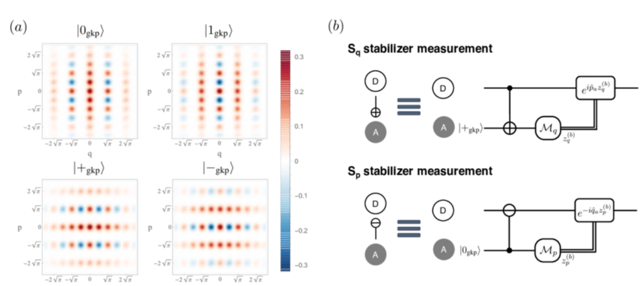


Figure 4: Logical $|0\rangle$, $|1\rangle$, $|+\rangle$, and $|-\rangle$ states in the non-ideal GKP code, along with corresponding syndrome measurement circuits.

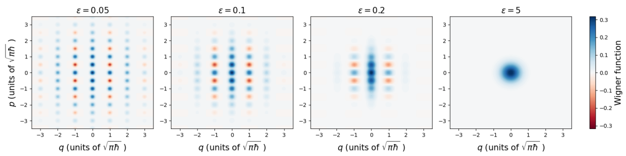


Figure 5: As we decrease the width of our envelope, the GKP codewords approach a Gaussian and perform worse.

It's possible to carry out Clifford gates on the logical qubits encoded in the GKP code through symplectic operations. These operations can be implemented through a combination of linear optics and squeezing, both of which are Gaussian operations.

The most difficult aspect of the GKP code is encoding, since this is the only part of the code which cannot be accomplished using only Gaussian states and operations. The GKP codewords themselves are not Gaussian. The initialization scheme proposed in the original paper is to prepare a squeezed state with $p = 0$ (as little uncertainty in p as possible) and then measure q modulo $\alpha = \sqrt{\pi}$. This modulo measurement is achieved by coupling the squeezed state for a short duration to another oscillator which serves as a meter.

3.3 Hexagonal GKP Code

The performance of the GKP code can be greatly improved by making it hexagonal. This requires syndrome measurements in three quadratures rather than two. The reason this improves the code is that with hexagonal symmetry, a larger fraction of potential error displacements are within the maximum correctable distance.

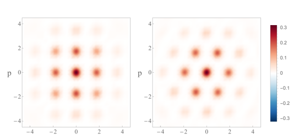


Figure 6: The maximally mixed state encoded in the GKP code (left) and hexagonal GKP code (right). [NAJ19]

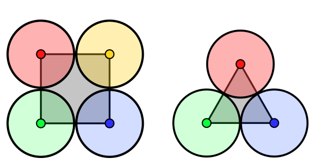


Figure 7: The Hexagonal GKP Code minimizes the probability an error state is beyond the maximum correctable displacement.

3.4 Applications of the GKP Code

In 2006, Nielsen et. al. devised CV cluster states that would allow for CV Measurement-Based Quantum Computation (MBQC). These relied on the GKP codestates as magic states: a source of non-Gaussianity which allowed for quantum universality [Men. et. al. 06]. In 2021, Furusawa et. al. showed how to create these CV cluster states by using multiplexing to generate GKP codestates [Asa et. al. 21]. Multiplexing is a way to speed up postselection by running many instances of a probabilistic process in parallel in order to increase the chances of the desired outcome. In 2021, Dhand et. al. gave a blueprint for scalable photonic Fault-Tolerant Quantum Computation (FTQC) by circumventing the exponential cost of postselection by replacing some proportion of GKP states in the cluster state with more easily prepared squeezed states. This was the ressult that eventually led to Xanadu's Borealis quantum computer, a fully programmable 216-qubit photonic quantum computer with quantum advantage.

In a related line of research, Menicucci et. al. showed in 2019 that it's possible to achieve universality with nothing but Gaussian elements. Their main result is worth quoting:

> Our main result is that a magic state for logical Clifford QC (using a particular Bosonic code) can be found within Gaussian QC. Thus, the union of these two simulable subtheories is universal and—with low enough physical noise—fault tolerant [BPAKM19].

The idea is to use GKP codes on Gaussian initial states (rather than the ideal GKP codewords), and use the fact that these states can be used as logical magic states in the GKP encoding.

References

[AF] Victor V. Albert, Philippe Faist. The Error Correction Zoo. https://errorcorrectionzoo.org/

[Ham50] R. W. Hamming. Error detecting and error correcting codes. *Bell Sys. Tech. Jou.*, 29,2: 147-160, 1950.

[Sho95] P. W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A.*, 52:R2493, 1995.

[CKN20] Jinkyu Cho, Young-Sik Kim, Jong-Seon No. Homomorphic Computation in Reed-Muller Codes. *IEEE*, Access 8: 108622-108628, 2020.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. *STOC*, p. 169-178, 2009.

[KLZ98] Emanuel Knill, Raymond Laflamme and Wojciech H. Zurek. Resilient quantum computation: error models and thresholds. *Proc. R. Soc. Lond.*, A.454365–384, 1998.

[Got97] Daniel Gottesman. Stabilizer Codes and Quantum Error Correction. Caltech Ph.D. Thesis, 1997.

[Got10] Daniel Gottesman. An Introduction to Quantum Error Correction. *Amer. Math. Soc.*, 68: p. 13-58, 2010.

[CLY97] I. L. Chuang, D. W. Leung, and Y. Yamamoto. Bosonic quantum codes for amplitude damping. *Phys. Rev. A*, 56,1114, 1997.

[WB07] W. Wasilewski and K. Banaszek. Protecting an optical qubit against photon loss. *Phys. Rev. A*, 75, 2007.

[Mich16] M. H. Michael et al. New Class of Quantum Error-Correcting Codes for a Bosonic Mode *Phys. Rev. X* 6, 2016.

[CYWZS21] W. Cai, M. Yuwei, W. Wang, C.-L. Zou, and L. Sun. Bosonic quantum error correction codes in superconducting quantum circuits. *Fundamental Research.* 1. 10.1016/j.fmre.2020.12.006. 2021.

[TCV20] B. M. Terhal and J. Conrad and C. Vuillot. Towards scalable Bosonic quantum error correction. *Quantum Sci. Technol.*, 5 043001, 2020.

[ME12] A. Mari and J. Eisert. Positive Wigner Functions Render Classical Simulation of Quantum Computation Efficient *Phys. Rev. Lett.* 109, 230503, 2012.

[GKP01] Daniel Gottesman, Alexei Y. Kitaev and John Preskill. Encoding a qubit in an oscillator. *Phys. Rev. A*, 64: 012310, 2001.

[Wang17] Yang Wang. Quantum Error Correction with the GKP Code and Concatenation with Stabilizer Codes. RWTH Aachen Ph.D. Thesis, 2017.

[NAJ19] K. Noh, V. V. Albert and L. Jiang. Quantum Capacity Bounds of Gaussian Thermal Loss Channels and Achievable Rates With Gottesman-Kitaev-Preskill Codes *IEEE*, 65,4:2563-2582, 2019.

[Men. et. al. 06] Nicolas Menicucci, Peter Loock, Mile Gu, Christian Weedbrook, Timothy Ralph, and Michael Nielsen. Universal Quantum Computation with Continuous-Variable Cluster States. *Phys. rev. lett.* 97:110501, 2006.

[BPAKM19] Ben Q. Baragiola, Giacomo Pantaleoni, Rafael N. Alexander, Angela Karanjai, and Nicolas C. Menicucci. All-Gaussian universality and fault tolerance with the Gottesman-Kitaev-Preskill code. *Phys. Rev. Lett.* 123:200502, 2019.

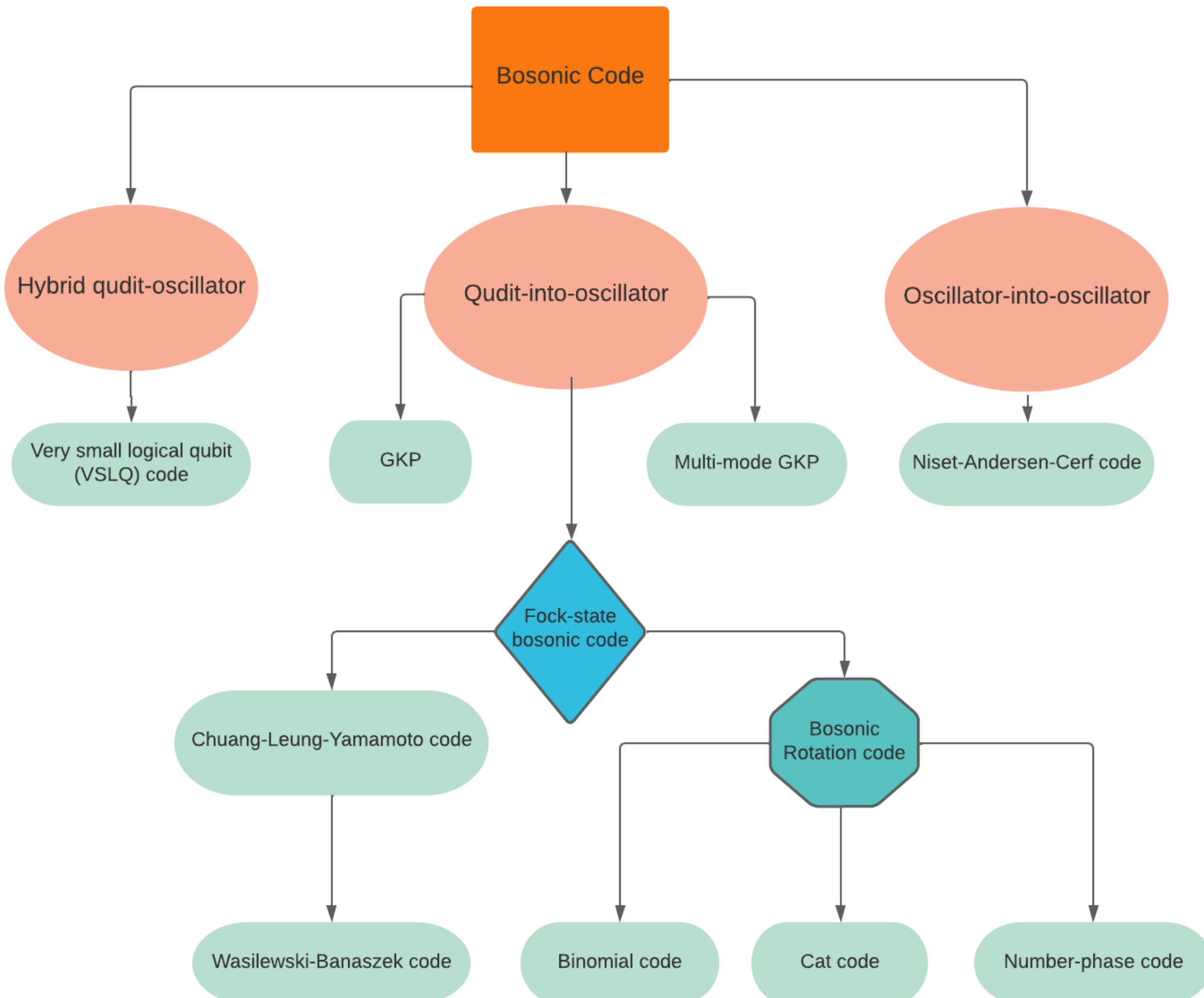
[Asa et. al. 21] Warit Asavanant, Yu Shiozawa, Shota Yokoyama, Baramee Charoensombutamon, Hiroki Emura, Rafael N. Alexander, Shuntaro Takeda, Jun-ichi Yoshikawa, Nicolas C. Menicucci, Hidehiro Yonezawa, and Akira Furusawa. Generation of time-domain-multiplexed two-dimensional cluster state. *Science* 366,6463:373-376, 2019.

[Bou et. al. 21] J. Eli Bourassa, Rafael N. Alexander, Michael Vasmer, Ashlesha Patil, Ilan Tzitrin, Takaya Matsuura, Daiqin Su, Ben Q. Baragiola, Saikat Guha, Guillaume Dauphinais, Krishna K. Sabapathy, Nicolas C. Menicucci, Ish Dhand. Blueprint for a Scalable Photonic Fault-Tolerant Quantum Computer. *Quantum* 5: 392, 2021.

[Mad et. al. 21] Lars S. Madsen, Fabian Laudenbach, Mohsen Falamarzi. Askarani, Fabien Rortais, Trevor Vincent, Jacob F. F. Bulmer, Filippo M. Miatto, Leonhard Neuhaus, Lukas G. Helt, Matthew J. Collins, Adriana E. Lita, Thomas Gerrits, Sae Woo Nam, Varun D. Vaidya, Matteo Menotti, Ish Dhand, Zachary Vernon, Nicolás Quesada, and Jonathan Lavoie. Quantum computational advantage with a programmable photonic processor. *Nature* 606: 75–81, 2022.

[Case08] William B. Case. Wigner functions and Weyl transforms for pedestrians. *American Journal of Physics* 76, 937-946, 2008.

A Tree Diagram of Bosonic Code Families



B Wigner Functions [Case08]

We can express any state $\psi(x)$ in the momentum basis as

$$\varphi(p) = \frac{1}{\sqrt{\hbar}} \int e^{-ixp/\hbar} \psi(x)\, dx = \langle p|\psi\rangle.$$

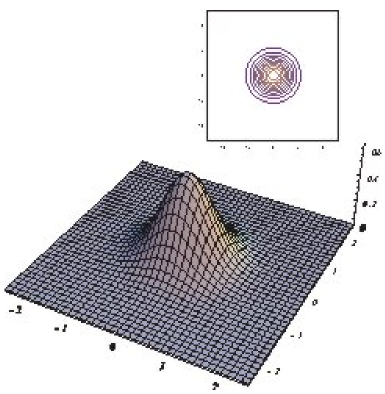
This is just a Fourier transform. The Wigner function of ψ is defined as

$$W(x,p) = \frac{1}{h} \int e^{-ipy/\hbar} \psi(x + y/2)\psi^*(x - y/2)\, dy.$$

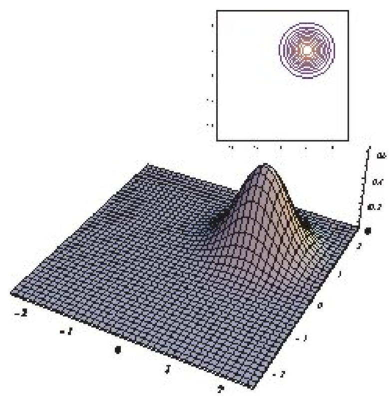
This function was defined to satisfy the properties

$$\int W(x,p)\, dp = \psi^*(x)\psi(x) \qquad \text{and} \qquad \int W(x,p)\, dx = \varphi^*(p)\varphi(p).$$

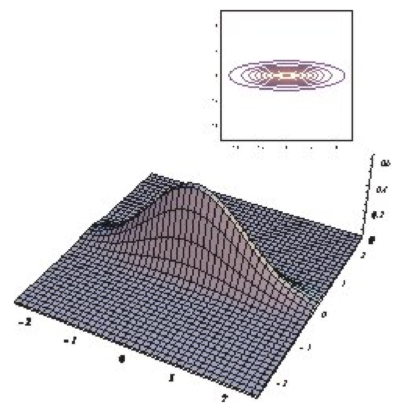
Gaussian States



Vacuum State

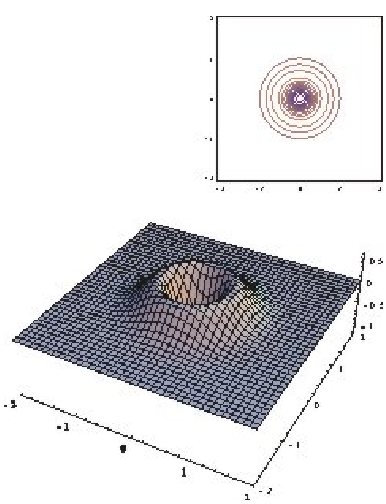


Coherent State

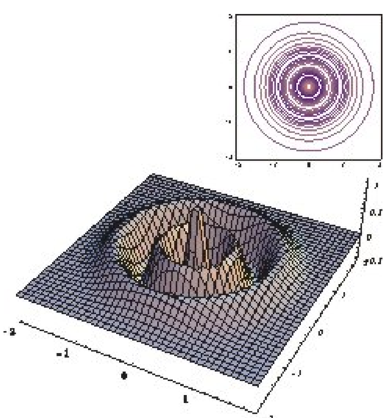


Squeezed State

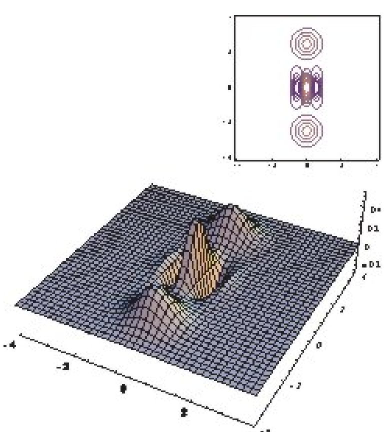
Non-Gaussian States



1-photon Fock State



4-photon Fock State



$|\Psi\rangle = |\alpha\rangle + |-\alpha\rangle$

Cat State

Image source: `http://old.rqc.ru/quantech/wiggalery.php`