Supplemental Information for “Neural Network-Based Design of Optimal Approximate GKP Codes“

Yexiong Zeng,[1, 2] Wei Qin,[3, 1, 4, *] Ye-Hong Chen,[5, 6, 1, 2] Clemens Gneiting,[1, 2, †] and Franco Nori[1, 2, 7, ‡]

[1] *Theoretical Quantum Physics Laboratory, Cluster for Pioneering Research, RIKEN, Wakoshi, Saitama 351-0198, Japan*
[2] *Quantum Computing Center, RIKEN, Wakoshi, Saitama 351-0198, Japan*
[3] *Center for Joint Quantum Studies and Department of Physics,*
School of Science, Tianjin University, Tianjin 300350, China
[4] *Tianjin Key Laboratory of Low Dimensional Materials Physics and*
Preparing Technology, Tianjin University, Tianjin 300350, China
[5] *Fujian Key Laboratory of Quantum Information and Quantum Optics, Fuzhou University, Fuzhou 350116, China*
[6] *Department of Physics, Fuzhou University, Fuzhou 350116, China*
[7] *Department of Physics, University of Michigan, Ann Arbor, Michigan, 48109-1040, USA*

CONTENTS

I. DEFINING THE APPROXIMATE SQUARE GKP CODEWORDS

Here, we give the form of the approximate GKP codewords and analyze their features. Specifically, the approximate GKP state can be expanded as a superposition of squeezed coherent states [1]

$$|u_L\rangle = \frac{1}{\mathcal{N}(u)} \sum_{k=-M}^{M} c_k^{(u)} \left|\alpha_k^{(u)}, r\right\rangle, \quad u \in \{0,\ 1\}, \tag{S1}$$

where $\left|\alpha_k^{(u)}, r\right\rangle = \hat{D}(\alpha_k^{(u)})\hat{S}(r)|0\rangle$ is a squeezed coherent state with the displacement $\hat{D}(\alpha) = \exp\left(\alpha\hat{a}^\dagger - \alpha^*\hat{a}\right)$ and squeezing $\hat{S}(r) = \exp\left[\frac{1}{2}\left(r^*\hat{a}^2 - r\hat{a}^{\dagger 2}\right)\right]$ operators, $\mathcal{N}(u)$ is the coefficient for normalization, and the coefficients $c_k^{(u)}$ are functions of the parameters $[\beta_k^{(u)}, u, k]$ [i.e., $c_k^{(u)} = f(\beta_k^{(u)},\ k,\ u)$] with the two-photon coherent parameter $\beta_k^{(u)} = \cosh(r)\alpha_k^{(u)} + \sinh(r)\alpha_k^{(u)*}$ [2]. Specifically, we rewrite the squeezed coherent state as $|\alpha_k^{(u)}, r\rangle = \hat{S}(r)\hat{S}(-r)\hat{D}(\alpha_k^{(u)})\hat{S}(r)|0\rangle = \hat{S}(r)\hat{D}(\beta_k^{(u)})|0\rangle$, which is the eigenstate of the squeezed mode $\hat{b} = \hat{S}(r)\hat{a}\hat{S}(-r)$ with eigenvalue $\beta_k^{(u)}$. This allows us to calculate analytical solutions for the expectation values of operators in the squeezing frame and avoids numerical truncations, that is, $\alpha_k^{(u)} \to \beta_k^{(u)}$ and $\hat{a} \to \hat{b}$. Here, we focus on the square GKP codewords and assume a real squeezing parameter r. According to Eq. (S1), we can express the normalization coefficient $\mathcal{N}(u)$

[*] qin.wei@tju.edu.cn
[†] clemens.gneiting@riken.jp
[‡] fnori@riken.jp

as:

$$\begin{aligned}\mathcal{N}^2(u) &= \sum_{k,l=0} f^*\left(\beta_k^{(u)},k,u\right) f\left(\beta_l^{(u)},l,u\right)\left\langle \alpha_k^{(u)},r|\alpha_l^{(u)},r\right\rangle \\ &= \sum_{k,l=0} f^*\left(\beta_k^{(u)},k,u\right) f\left(\beta_l^{(u)},l,u\right)\exp\left[-\frac{\left|\beta_k^{(u)}\right|^2+\left|\beta_l^{(u)}\right|^2}{2}+\beta_k^{(u)*}\beta_l^{(u)}\right].\end{aligned} \tag{S2}$$

The codewords in Eq. (S1) can be regarded as the shared approximate eigenstates of the two stabilizer operators of the square GKP code

$$\hat{S}_q = \exp\left(i2\sqrt{\pi}\hat{q}\right) = \hat{D}\left(\sqrt{2\pi}i\right), \qquad \hat{S}_p = \exp\left(-i2\sqrt{\pi}\hat{p}\right) = \hat{D}\left(\sqrt{2\pi}\right), \tag{S3}$$

with the same eigenvalue one, where $\hat{q} = (\hat{a}^\dagger + \hat{a})/\sqrt{2}$ and $\hat{p} = (\hat{a} - \hat{a}^\dagger)/\sqrt{2}i$ represent the position and momentum operators, respectively. Using the relation $\hat{D}(\beta)\hat{D}(\alpha) = \exp\left[(\beta\alpha^* - \beta^*\alpha)/2\right]\hat{D}(\alpha+\beta)$, we can derive the following expressions:

$$\begin{aligned}\hat{S}_q\left|\alpha_l^{(u)},r\right\rangle &= \hat{D}\left(\sqrt{2\pi}i\right)\hat{D}\left(\alpha_l^{(u)}\right)\hat{S}\left(r\right)|0\rangle = \exp\left[\frac{\sqrt{2\pi}i}{2}\left(\alpha_l^{(u)*}+\alpha_l^{(u)}\right)\right]\left|\alpha_l^{(u)}+\sqrt{2\pi}i,r\right\rangle, \\ \hat{S}_p\left|\alpha_l^{(u)},r\right\rangle &= \hat{D}\left(\sqrt{2\pi}\right)\hat{D}\left(\alpha_l^{(u)}\right)S\left(r\right)|0\rangle = \exp\left[\frac{\sqrt{2\pi}}{2}\left(\alpha_l^{(u)*}-\alpha_l^{(u)}\right)\right]\left|\alpha_l^{(u)}+\sqrt{2\pi},r\right\rangle.\end{aligned} \tag{S4}$$

By combining the Eqs. (S1) and (S4), we obtain the inner products

$$\begin{aligned}\left\langle u_L\left|\hat{S}_q\right|u_L\right\rangle &= \exp\left(-\pi e^{-2r}\right), \\ \left\langle u_L\left|\hat{S}_p\right|u_L\right\rangle &= \frac{1}{\mathcal{N}^2(u)}\sum_{k,l=0} f^*\left(\beta_k^{(u)},k,u\right) f\left(\beta_l^{(u)},l,u\right)\exp\left[-\frac{\left|\beta_k^{(u)}\right|^2+\left|\beta_l^{(u)'}\right|^2}{2}+\beta_k^{(u)*}\beta_l^{(u)'}\right],\end{aligned} \tag{S5}$$

where we have defined the relation $\beta_l^{(u)'} = \beta_l^{(u)} + \sqrt{2\pi}e^r$ and $\left(\alpha_k^{(u)}\sqrt{\frac{2}{\pi}} - u\right)/2 \in \mathbb{Z}$. Note that we focus on the most general form of the GKP code, more specifically the *square* GKP code, whose Wigner function is a square grid. The positions of the grid points in this code are determined by the parameters $\alpha_k^{(u)}$. Thus, we have fixed these parameters to ensure a precise approximation and to maintain the consistency with the ideal square GKP code. The difference between the approximate GKP codewords and the eigenstates of the operator $\hat{S}_q$ [i.e., the right side of the first line in Eq. (S5)] depends solely on the squeezing parameter r. However, the gap between the approximate GKP codewords and the eigenstate of the operator $\hat{S}_p$ is determined by the squeezing strength r and the coefficients $c_k^{(u)}$ of the squeezed coherent states. These results provide us with the potential to obtain an optimal code. Since the distance between the *ideal* GKP code and the *approximate* GKP code is determined by the squeezing strength, and the level of approximation along the $\hat{S}_q$ direction is fixed, we only need to ensure that $\hat{S}_p$ meets or exceeds this level of approximation to qualify as a square GKP code. Therefore, we strive to keep $\langle u_L|\hat{S}_p|u_L\rangle \geqslant \exp\left(-\pi e^{-2r}\right)$, thereby ensuring a good approximation to the ideal codewords. To achieve this, we incorporate the following component into the loss function:

$$L_{\text{eg}} = \sum_{u=0,1}\max\left(0,\exp\left(-\pi e^{-2r}\right) - \left\langle u_L\left|\hat{S}_p\right|u_L\right\rangle\right) \tag{S6}$$

II. QUANTUM ERROR CORRECTION ABILITY

This section provides a detailed calculation of the QEC ability. The primary source of errors in a bosonic mode are single-photon loss and dephasing. Hence, the system dynamics is governed by the Lindblad master equation

$$\frac{\mathrm{d}\hat{\rho}}{\mathrm{dt}} = \frac{\kappa}{2}\mathcal{D}\left[\hat{a}\right] + \frac{\kappa_\phi}{2}\mathcal{D}\left[\hat{a}^\dagger\hat{a}\right], \quad \mathcal{D}\left[\hat{x}\right] = 2\hat{x}\hat{\rho}\hat{x}^\dagger - \hat{x}^\dagger\hat{x}\hat{\rho} - \hat{\rho}\hat{x}^\dagger\hat{x}. \tag{S7}$$

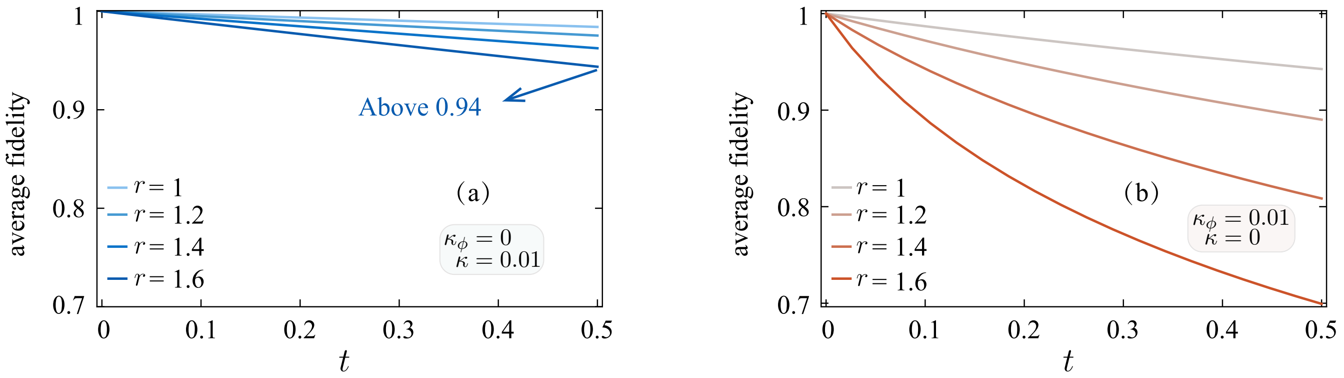

FIG. S1. The average fidelity $\bar{F}$ of the logical code space versus time t for various squeezing strengths: (a) $\kappa_\phi = 0$ and $\kappa = 0.01$; (b) $\kappa_\phi = 0.01$ and $\kappa = 0$.

To demonstrate the effects of dephasing and single-photon loss, we evaluate the evolution of the average fidelity for the conventional GKP codespace, as shown in Fig. S1. The average fidelity is $\bar{F} = \frac{1}{6}\sum_i \mathrm{Tr}[\hat{\rho}_i \mathcal{N}(\hat{\rho}_i)]$, where $\mathcal{N}(\cdot)$ represents the dynamical map governed by the master equation in Eq. (S7) and $\hat{\rho}_i$ denotes the six Pauli eigenstates of the logical space. The impact of the dephasing channel becomes more pronounced with increasing the degree of squeezing. To elucidate the effect of dephasing in detail, we provide an analytical solution for the dephasing channel in the Fock space

$$\mathcal{N}_\phi(\rho) = \sum_{nm} \exp\left\{\left[-\frac{\kappa_\phi}{2}(n-m)^2\right]\right\}\langle n|\rho|m\rangle|n\rangle\langle m|, \tag{S8}$$

where the dynamic map $\mathcal{N}_\phi(\cdot)$ describes the pure dephasing process. As the squeezing degree increases, the distribution of codewords in the Fock space becomes more dispersed (i.e., the magnitude of $|n-m|$ increases), which exacerbates the detrimental effects of dephasing.

By considering short times $\kappa\tau$, $\kappa_\phi\tau \ll 1$, we can expand the time-dependent density matrix $\rho(\tau)$ by using the Kraus operators $\hat{A}_k(\tau)$ [3]

$$\begin{gathered}\hat{\rho}(\tau) = \sum_k \hat{A}_k(\tau)\hat{\rho}(0)\hat{A}_k(\tau)^\dagger + \mathcal{O}(\tau^2),\\ \hat{A}_1(\tau) = \sqrt{\hat{I} - \kappa\tau\hat{a}^\dagger\hat{a} - \kappa_\phi\tau(\hat{a}^\dagger\hat{a})^2} \approx \hat{I} - \frac{\kappa\tau}{2}\hat{a}^\dagger\hat{a} - \frac{\kappa_\phi\tau}{2}(\hat{a}^\dagger\hat{a})^2, \qquad \hat{A}_2(\tau) = \sqrt{\kappa\tau}\hat{a}, \qquad \hat{A}_3(\tau) = \sqrt{\kappa_\phi\tau}\hat{a}^\dagger\hat{a}.\end{gathered} \tag{S9}$$

We consider an approximate QEC characterized by a finite squeezing amplitude r and the form of the codewords; that is, the codewords satisfies approximately the Knill-Laflamme (KL) condition

$$\hat{P}_C\hat{A}_i^\dagger\hat{A}_j\hat{P}_C = T_{ij}\hat{P}_c + \hat{\Delta}_{ij}, \tag{S10}$$

where $\hat{P}_C$ is the projector onto the code space, T_{ij} are the elements of a Hermitian matrix T, and $\hat{\Delta}_{ij}$ is a residual error operator. We use the eigendecomposition of the T matrix, $T = U\Lambda U^\dagger$, and then remodel the Kraus operators as $\hat{F}_i = \sum_k U_{ki}\hat{A}_k$, resulting in

$$\sum_i \hat{F}_i\hat{\rho}\hat{F}_i^\dagger = \sum_{kl}\sum_i U_{ki}U_{li}^*\hat{A}_k\hat{\rho}\hat{A}_l^\dagger = \sum_k \hat{A}_k\hat{\rho}\hat{A}_k^\dagger. \tag{S11}$$

With this, we can further expand the KL condition as [4]

$$\hat{P}_C\hat{F}_i^\dagger\hat{F}_j\hat{P}_C = \sum_{kl} U_{ki}^*U_{lj}\hat{P}_C\hat{A}_k^\dagger\hat{A}_l\hat{P}_C = \sum_{kl} U_{ki}^*U_{lj}(T_{kl}\hat{P}_C + \hat{\Delta}_{kl}) = \delta_{ij}\Lambda_{jj}\hat{P}_C + \hat{\tilde{\Delta}}_{ij}, \tag{S12}$$

where we have defined $\hat{\tilde{\Delta}}_{ij} = \sum_{ki} U_{ki}^*U_{lj}\hat{\Delta}_{kl}$.

We can employ the recovery operator $\hat{R}_j$ to correct the error resulting from $\hat{A}_i(\tau)$. The recovery operator is not unique. If we consider the recovery operator $\hat{R}_i = \hat{P}_C\hat{F}_i^\dagger/\sqrt{\Lambda_{ii}}$ and $\hat{R}_0 = \sqrt{\hat{I} - \sum_i \hat{R}_i^\dagger\hat{R}_i}$ to correct the errors, we obtain the recovery channel

$$\mathcal{E}(\hat{\rho}) = \mathcal{R}\circ\mathcal{N}(\hat{\rho}) = \sum_{ij}\frac{1}{\Lambda_{ii}}\hat{P}\hat{F}_i^{\dagger}\hat{F}_j\hat{\rho}\hat{F}_j^{\dagger}\hat{F}_i\hat{P} = \sum_i \Lambda_{ii}\hat{\rho} + \sum_{ij}\hat{\hat{\Delta}}_{ij}\hat{\rho} + \hat{\rho}\hat{\hat{\Delta}}_{ji} + \frac{1}{\Lambda_{jj}}\hat{\hat{\Delta}}_{ij}\hat{\hat{\Delta}}_{ji} + \hat{R}_0\mathcal{N}(\hat{\rho})\hat{R}_0^{\dagger}, \tag{S13}$$

which indicates that if the KL criteria is satisfied (i.e., $\hat{\Delta}_{ij} = \hat{\hat{\Delta}}_{ij} \to 0$), the information can be corrected fully [i.e., $\mathcal{R}(\hat{\rho}) = \mathcal{N}(\hat{\rho}) = \hat{\rho}$].

Hence, we aim to bring the approximate GKP codewords as close as possible to the KL condition

$$\epsilon_{ji} = \langle 1_L|\hat{A}_j^{\dagger}\hat{A}_i|1_L\rangle - \langle 0_L|\hat{A}_j^{\dagger}\hat{A}_i|0_L\rangle,\ \zeta_{ji} = \langle 0_L|\hat{A}_j^{\dagger}\hat{A}_i|1_L\rangle,\ \zeta_{ij}^{*} = \langle 1_L|\hat{A}_j^{\dagger}\hat{A}_i|0_L\rangle,\ \delta = \langle 0_L|1_L\rangle. \tag{S14}$$

It should be noted that the above recovery may not be the optimal solution. We can address this by solving convex optimization problems to obtain the optimal channel fidelity [5–7]. To ensure that the logical basis vectors meet the KL criteria as close as possible, we define the following loss function

$$\bar{L}_{\mathrm{er}} = \frac{1}{N}\sum_{\kappa\tau,\kappa_{\phi}\tau} L_{\mathrm{er}}, \tag{S15}$$

where we have considered L_{er} as

$$L_{\mathrm{er}} = |\delta| + \sum_{ij}\left(|\epsilon_{ji}| + |\zeta_{ij}|\right). \tag{S16}$$

Here N is the number of terms summed over in $\bar{L}_{\mathrm{er}}$ to guarantee the loss function can be sufficiently minimal. Moreover, we calculate the following relation

$$\hat{P}_C\hat{A}_j^{\dagger}\hat{A}_i\hat{P}_C = \langle 0_L|\hat{A}_j^{\dagger}\hat{A}_i|0_L\rangle\hat{P}_C + \epsilon_{ji}|1_L\rangle\langle 1_L| + \zeta_{ij}^{*}|1_L\rangle\langle 0_L| + \zeta_{ji}|0_L\rangle\langle 1_L|, \tag{S17}$$

which implies that ϵ_{ji} and ζ_{ji} describe the Pauli $\hat{\sigma}_z$ errors and also $\hat{\sigma}_x$ or $\hat{\sigma}_y$ errors, respectively. If ζ_{ji} and ϵ_{ji} vanish, the KL conditions are satisfied (i.e., no noise bias). These noise biases are reduced roughly equally during the optimization process due to the consistent weights of errors in the loss function, ensuring that the final noise bias reaches small values.

Moreover, the Eq. (S14) can be expressed as the sum of $\langle u_L|\hat{K}_i^{\dagger}\hat{K}_j|v_L\rangle$, where $\hat{K}_i$ and $\hat{K}_j$ are operators selected from the set $\{\hat{I},\ \hat{a},\ \hat{a}^{\dagger}\hat{a},\ \hat{a}^{\dagger}\hat{a}\hat{a}^{\dagger}\hat{a}\}$ and u, v are binary values (i.e., $u, v \in \{0,\ 1\}$). Since numerical truncation can be computationally expensive and memory-intensive, we perform an analytical calculation of the parameter $\langle u_L|\hat{K}_i^{\dagger}\hat{K}_j|v_L\rangle$ to simplify the numerical simulation. We can expand $\langle u_L|\hat{K}_i^{\dagger}\hat{K}_j|v_L\rangle$ into the form

$$\begin{aligned}\left\langle u_L\left|\hat{K}_i^{\dagger}\hat{K}_j\right|v_L\right\rangle =& \frac{1}{\mathcal{N}(u)\mathcal{N}(v)}\sum_{k,l=0} f^{*}\left(\alpha_k^{(u)}, r\right) f\left(\alpha_l^{(v)}, r\right)\left\langle \alpha_k^{(u)}, r\left|\hat{K}_i^{\dagger}\hat{K}_j\right|\alpha_k^{(v)}, r\right\rangle \\ =& \frac{1}{\mathcal{N}(u)\mathcal{N}(v)}\sum_{k,l=0} f^{*}\left(\alpha_k^{(u)}, r\right) f\left(\alpha_l^{(v)}, r\right) G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right)\exp\left[-\frac{\left|\beta_k^{(u)}\right|^2 + \left|\beta_l^{(v)}\right|^2}{2} + \beta_k^{(u)*}\beta_l^{(v)}\right].\end{aligned} \tag{S18}$$

The following is an analytic formulation of $G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right)$. We have assumed $\lambda = \cosh(r)$ and $\lambda_1 = \sinh(r)$ for more concise expressions.

1. For the operator $\hat{K}_i^{\dagger}\hat{K}_j = \hat{I}$, we obtain

$$G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) = 1; \tag{S19}$$

2. For the operator $\hat{K}_i^{\dagger}\hat{K}_j = \hat{a}$, we obtain

$$G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) = \lambda\beta_l^{(v)} - \lambda_1\beta_k^{(u)*}; \tag{S20}$$

3. For the operator $\hat{K}_i^{\dagger}\hat{K}_j = \hat{a}^{\dagger}\hat{a}$, we obtain

$$G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) = \beta_l^{(v)}\beta_k^{(u)*}\left(\lambda^2 + \lambda_1^2\right) + \lambda_1\left[\lambda_1 - \lambda\left(\beta_l^{(v)}\right)^2\right] - \lambda\lambda_1\left(\beta_k^{(u)*}\right)^2; \tag{S21}$$

4. For the operator $\hat{K}_i^\dagger \hat{K}_j = \hat{a}^\dagger \hat{a}^2$, we obtain

$$\begin{aligned} G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) =& -\lambda_1\left(2\lambda^2+\lambda_1^2\right)\left(\beta_k^{(u)*}\right)^2 \beta_l^{(v)} + \left(\lambda^2+2\lambda_1^2\right)\beta_k^{(u)*}\left[\lambda\left(\beta_l^{(v)}\right)^2-\lambda_1\right] + \lambda\lambda_1^2\left(\beta_k^{(u)*}\right)^3 \\ & + \lambda\lambda_1\beta_l^{(v)}\left[3\lambda_1-\lambda\left(\beta_l^{(v)}\right)^2\right]; \end{aligned} \tag{S22}$$

5. For the operator $\hat{K}_i^\dagger \hat{K}_j = (\hat{a}^\dagger \hat{a})^2$, we obtain

$$\begin{aligned} G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) =& -2\lambda\lambda_1\left(\lambda^2+\lambda_1^2\right)\left(\beta_k^{(u)*}\right)^3\beta_l^{(v)} + \left(\beta_k^{(u)*}\right)^2\left[\lambda^4\left(\beta_l^{(v)}\right)^2 + 4\lambda_1^2\lambda^2\left(\beta_l^{(v)}\right)^2 - 2\lambda_1\lambda^3 - 4\lambda_1^3\lambda\right. \\ & \left. +\lambda_1^4\left(\beta_l^{(v)}\right)^2\right] + \beta_k^{(u)*}\beta_l^{(v)}\left[-2\lambda_1\lambda^3\left(\beta_l^{(v)}\right)^2 - 2\lambda_1^3\lambda\left(\beta_l^{(v)}\right)^2 + \lambda^4 + 8\lambda_1^2\lambda^2 + 3\lambda_1^4\right] \\ & + \lambda^2\lambda_1^2\left(\beta_k^{(u)*}\right)^4 + \lambda_1\left\{-2\lambda^3\left(\beta_l^{(v)}\right)^2 + \lambda_1\lambda^2\left[\left(\beta_l^{(v)}\right)^4+2\right] - 4\lambda_1^2\lambda\left(\beta_l^{(v)}\right)^2 + \lambda_1^3\right\}; \end{aligned} \tag{S23}$$

6. For the operator $\hat{K}_i^\dagger \hat{K}_j = \left(\hat{a}^\dagger \hat{a}\right)^2 \hat{a}$, we obtain

$$\begin{aligned} G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) =& \lambda\lambda_1^2\left(3\lambda^2+2\lambda_1^2\right)\left(\beta_k^{(u)*}\right)^4\beta_l^{(v)} - \lambda_1\left(\beta_k^{(u)*}\right)^3\left[3\lambda^4\left(\beta_l^{(v)}\right)^2 + 6\lambda_1^2\lambda^2\left(\beta_l^{(v)}\right)^2 - 4\lambda_1\lambda^3\right. \\ & \left. +\lambda_1^4\left(\beta_l^{(v)}\right)^2 - 6\lambda_1^3\lambda\right] + \left(\beta_k^{(u)*}\right)^2\beta_l^{(v)}\left[\lambda^5\left(\beta_l^{(v)}\right)^2 + 3\lambda_1^4\lambda\left(\beta_l^{(v)}\right)^2 - 5\lambda_1\lambda^4 - 20\lambda_1^3\lambda^2\right. \\ & \left. +6\lambda_1^2\lambda^3\left(\beta_l^{(v)}\right)^2 - 5\lambda_1^5\right] + \beta_k^{(u)*}\left\{\lambda^5\left(\beta_l^{(v)}\right)^2 - \lambda_1\lambda^4\left[2\left(\beta_l^{(v)}\right)^4+1\right] + 16\lambda_1^2\lambda^3\left(\beta_l^{(v)}\right)^2\right. \\ & \left. -\lambda_1^3\lambda^2\left[3\left(\beta_l^{(v)}\right)^4+10\right] + 13\lambda_1^4\lambda\left(\beta_l^{(v)}\right)^2 - 4\lambda_1^5\right\} - \lambda^2\lambda_1^3\left(\beta_k^{(u)*}\right)^5 \\ & + \lambda\lambda_1\beta_l^{(v)}\left\{-2\lambda^3\left(\beta_l^{(v)}\right)^2 + \lambda_1\lambda^2\left[\left(\beta_l^{(v)}\right)^4+6\right] - 8\lambda_1^2\lambda\left(\beta_l^{(v)}\right)^2 + 9\lambda_1^3\right\}; \end{aligned} \tag{S24}$$

7. For the operator $\hat{K}_i^\dagger \hat{K}_j = (\hat{a}^\dagger \hat{a})^3$, we obtain

$$\begin{aligned} G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) =& 3\lambda^2\lambda_1^2\left(\lambda^2+\lambda_1^2\right)\left(\beta_k^{(u)*}\right)^5\beta_l^{(v)} - 3\lambda\lambda_1\left(\beta_k^{(u)*}\right)^4\left[\lambda^4\left(\beta_l^{(v)}\right)^2 + 3\lambda_1^2\lambda^2\left(\beta_l^{(v)}\right)^2 - 2\lambda_1\lambda^3\right. \\ & \left. +\lambda_1^4\left(\beta_l^{(v)}\right)^2 - 3\lambda_1^3\lambda\right] + \left(\beta_k^{(u)*}\right)^3\beta_l^{(v)}\left[\lambda^6\left(\beta_l^{(v)}\right)^2 + 9\lambda_1^2\lambda^4\left(\beta_l^{(v)}\right)^2 + 9\lambda_1^4\lambda^2\left(\beta_l^{(v)}\right)^2\right. \\ & \left. +\lambda_1^6\left(\beta_l^{(v)}\right)^2 - 9\lambda_1\lambda^5 - 36\lambda_1^3\lambda^3 - 15\lambda_1^5\lambda\right] + \left(\beta_k^{(u)*}\right)^2\left\{3\lambda^6\left(\beta_l^{(v)}\right)^2 - \lambda_1\lambda^5\left[3\left(\beta_l^{(v)}\right)^4+4\right]\right. \\ & +6\lambda_1^6\left(\beta_l^{(v)}\right)^2 + 36\lambda_1^2\lambda^4\left(\beta_l^{(v)}\right)^2 - \lambda_1^3\lambda^3\left[9\left(\beta_l^{(v)}\right)^4+28\right] - \lambda_1^5\lambda\left[3\left(\beta_l^{(v)}\right)^4+13\right] \\ & \left. +45\lambda_1^4\lambda^2\left(\beta_l^{(v)}\right)^2\right\} + \beta_k^{(u)*}\beta_l^{(v)}\left\{-9\lambda_1\lambda^5\left(\beta_l^{(v)}\right)^2 + \lambda_1^2\lambda^4\left[3\left(\beta_l^{(v)}\right)^4+32\right]\right. \\ & \left. -36\lambda_1^3\lambda^3\left(\beta_l^{(v)}\right)^2 + \lambda_1^4\lambda^2\left[3\left(\beta_l^{(v)}\right)^4+50\right] - 15\lambda_1^5\lambda\left(\beta_l^{(v)}\right)^2 + \lambda^6 + 7\lambda_1^6\right\} - \lambda^3\lambda_1^3\left(\beta_k^{(u)*}\right)^6 \\ & + \lambda_1\left\{-4\lambda^5\left(\beta_l^{(v)}\right)^2 + 2\lambda_1\lambda^4\left[3\left(\beta_l^{(v)}\right)^4+2\right] - \lambda_1^2\lambda^3\left(\beta_l^{(v)}\right)^2\left[\left(\beta_l^{(v)}\right)^4+28\right]\right. \\ & \left. +\lambda_1^3\lambda^2\left[9\left(\beta_l^{(v)}\right)^4+10\right] - 13\lambda_1^4\lambda\left(\beta_l^{(v)}\right)^2 + \lambda_1^5\right\}; \end{aligned} \tag{S25}$$

8. For the operator $\hat{K}_i^\dagger \hat{K}_j = (\hat{a}^\dagger \hat{a})^4$, we obtain

$$\begin{aligned} G_j\left(\beta_k^{(u)}, \beta_l^{(v)}\right) =& \lambda^4\lambda_1^4\left(\beta_k^{(u)*}\right)^8 - 4\lambda^3\lambda_1^3\left(\lambda^2+\lambda_1^2\right)\beta_l^{(v)}\left(\beta_k^{(u)*}\right)^7 + 2\lambda^2\lambda_1^2\left[3\left(\beta_l^{(v)}\right)^2\lambda^4 + 8\lambda_1^2\left(\beta_l^{(v)}\right)^2\lambda^2\right. \\ & \left. -6\lambda_1\lambda^3 - 8\lambda_1^3\lambda + 3\lambda_1^4\left(\beta_l^{(v)}\right)^2\right]\left(\beta_k^{(u)*}\right)^6 - 2\lambda\lambda_1\beta_l^{(v)}\left[2\left(\beta_l^{(v)}\right)^2\lambda^6 + 12\lambda_1^2\left(\beta_l^{(v)}\right)^2\lambda^4\right. \end{aligned}$$

$$\begin{aligned}
&-15\lambda_1\lambda^5-48\lambda_1^3\lambda^3+12\lambda_1^4\left(\beta_l^{(v)}\right)^2\lambda^2-21\lambda_1^5\lambda+2\lambda_1^6\left(\beta_l^{(v)}\right)^2\bigg]\left(\beta_k^{(u)*}\right)^5+\bigg\{\left(\beta_l^{(v)}\right)^4\lambda^8\\
&-192\lambda_1^5\left(\beta_l^{(v)}\right)^2\lambda^3-24\lambda_1\left(\beta_l^{(v)}\right)^2\lambda^7+4\lambda_1^4\left[9\left(\beta_l^{(v)}\right)^4+31\right]\lambda^4+\lambda_1^8\left(\beta_l^{(v)}\right)^4\\
&+4\lambda_1^2\left[4\left(\beta_l^{(v)}\right)^4+7\right]\lambda^6-168\lambda_1^3\left(\beta_l^{(v)}\right)^2\lambda^5+2\lambda_1^6\left[8\left(\beta_l^{(v)}\right)^4+29\right]\lambda^2\\
&-36\lambda_1^7\left(\beta_l^{(v)}\right)^2\lambda\bigg\}\left(\beta_k^{(u)*}\right)^4-2\beta_l^{(v)}\bigg\{2\lambda_1\left[\left(\beta_l^{(v)}\right)^4+8\right]\lambda^7+2\lambda_1^3\left[6\left(\beta_l^{(v)}\right)^4+79\right]\lambda^5\\
&-3\left(\beta_l^{(v)}\right)^2\lambda^8-144\lambda_1^4\left(\beta_l^{(v)}\right)^2\lambda^4+2\lambda_1^5\left[6\left(\beta_l^{(v)}\right)^4+103\right]\lambda^3+2\lambda_1^7\left[\left(\beta_l^{(v)}\right)^4+20\right]\lambda\\
&-56\lambda_1^2\left(\beta_l^{(v)}\right)^2\lambda^6-72\lambda_1^6\left(\beta_l^{(v)}\right)^2\lambda^2-5\lambda_1^8\left(\beta_l^{(v)}\right)^2\bigg\}\left(\beta_k^{(u)*}\right)^3+\bigg\{7\left(\beta_l^{(v)}\right)^2\lambda^8\\
&+2\lambda_1^2\left(\beta_l^{(v)}\right)^2\left[3\left(\beta_l^{(v)}\right)^4+106\right]\lambda^6-8\lambda_1\left[3\left(\beta_l^{(v)}\right)^4+1\right]\lambda^7-24\lambda_1^3\left[7\left(\beta_l^{(v)}\right)^4+6\right]\lambda^5\\
&+4\lambda_1^4\left(\beta_l^{(v)}\right)^2\left[4\left(\beta_l^{(v)}\right)^4+165\right]\lambda^4+2\lambda_1^6\left(\beta_l^{(v)}\right)^2\left[3\left(\beta_l^{(v)}\right)^4+178\right]\lambda^2+25\lambda_1^8\left(\beta_l^{(v)}\right)^2\\
&-4\lambda_1^7\left[9\left(\beta_l^{(v)}\right)^4+10\right]\lambda-12\lambda_1^5\left[16\left(\beta_l^{(v)}\right)^4+19\right]\lambda^3\bigg\}\left(\beta_k^{(u)*}\right)^2+\bigg\{-32\lambda_1\left(\beta_l^{(v)}\right)^2\lambda^7\\
&+6\lambda_1^2\left[5\left(\beta_l^{(v)}\right)^4+18\right]\lambda^6+24\lambda_1^4\left[4\left(\beta_l^{(v)}\right)^4+19\right]\lambda^4-4\lambda_1^5\left(\beta_l^{(v)}\right)^2\left[\left(\beta_l^{(v)}\right)^4+103\right]\lambda^3\\
&+\lambda^8+15\lambda_1^8+2\lambda_1^6\left[21\left(\beta_l^{(v)}\right)^4+130\right]\lambda^2-4\lambda_1^3\left(\beta_l^{(v)}\right)^2\left[\left(\beta_l^{(v)}\right)^4+79\right]\lambda^5\\
&-80\lambda_1^7\left(\beta_l^{(v)}\right)^2\lambda\bigg\}\beta_l^{(v)}\beta_k^{(u)*}+\lambda_1\bigg\{-12\lambda_1^2\left(\beta_l^{(v)}\right)^2\left[\left(\beta_l^{(v)}\right)^4+12\right]\lambda^5-40\lambda_1^6\left(\beta_l^{(v)}\right)^2\lambda\\
&-8\left(\beta_l^{(v)}\right)^2\lambda^7+\lambda_1^3\left[\left(\beta_l^{(v)}\right)^8+124\left(\beta_l^{(v)}\right)^4+60\right]\lambda^4-4\lambda_1^4\left(\beta_l^{(v)}\right)^2\left[4\left(\beta_l^{(v)}\right)^4+57\right]\lambda^3\\
&+4\lambda_1\left[7\left(\beta_l^{(v)}\right)^4+2\right]\lambda^6+2\lambda_1^5\left[29\left(\beta_l^{(v)}\right)^4+18\right]\lambda^2+\lambda_1^7\bigg\}.
\end{aligned}\tag{S26}$$

III. THE OPTIMAL RECOVERY CHANNEL

Here we elaborate the numerical method to identify the optimal recovery channel. We assume that the error space consists of the bases $|i^{(u)}\rangle = A_i|u_L\rangle/\sqrt{\langle u_L|A_i^\dagger A_i|u_L\rangle}$, where $\hat{A}_i$ are the Kraus operators of the error channel and $|u_L\rangle$ is the logical basis. Therefore, the recovery operators are linear superpositions of the operators $\hat{B}_i$,

$$\hat{R}_k = \sum_i x_{k,i}\hat{B}_i, \tag{S27}$$

where we have defined the operator $\hat{B}_i \in \{|0_\mathrm{L}\rangle\langle i^{(u)}|,\ |1_\mathrm{L}\rangle\langle i^{(u)}|\}$ to restore the encoded information from the error space into the logical space. We can find the optimal coefficients $x_{k,i}$ to obtain the optimal recover operators $\hat{R}_k$. To this end, we need to maximize the entanglement fidelity by optimizing the coefficient $x_{k,i}$ [5–8],

$$F = \frac{1}{4}\sum_{ij}\left|\mathrm{Tr}\left\{\hat{R}_i\hat{A}_j\right\}\right|^2, \tag{S28}$$

which is equivalent to solving the following convex semidefinite program

$$X_\mathrm{opt} = \underset{X}{\mathrm{argmax}}\ \frac{1}{4}\,\mathrm{Tr}\left\{XW\right\}, \quad \text{with} \quad \sum_{ij} X_{ij}\hat{B}_i^\dagger\hat{B}_j = \hat{I}, \quad X \succcurlyeq 0, \tag{S29}$$

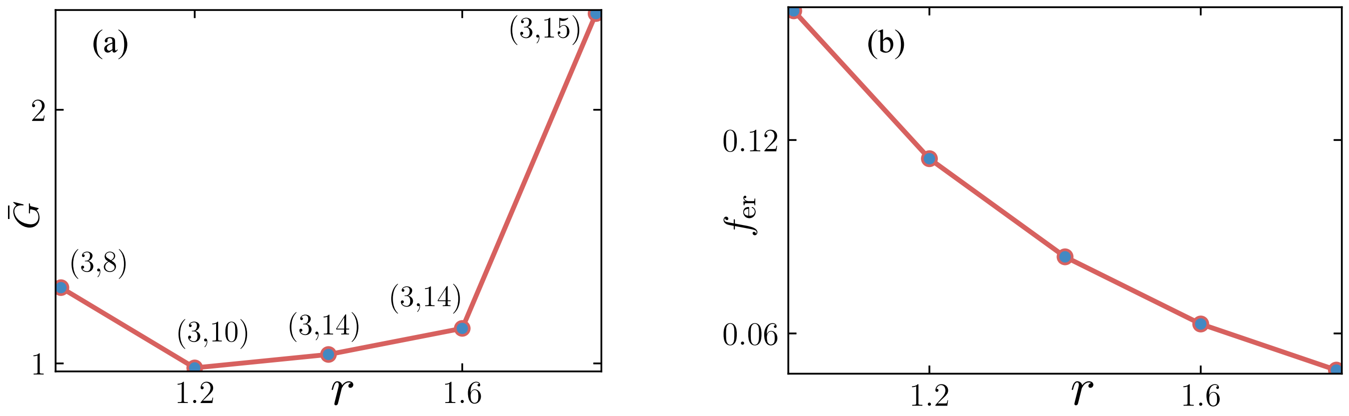

FIG. S2. (a) The gain of optimal GKP codewords relative to the best conventional GKP codewords versus the squeezing amplitude r. Here, (M_{o}, M_{c}) represents the fewest M values needed by the optimized GKP and the best conventional GKP codewords, respectively. The gains are above one for various squeezing amplitudes r, showing that the optimal GKP codewords significantly reduce the necessary M while maintaining higher error-correctability. (b) The distance between the stabilizer operators of the ideal and optimal approximate GKP codewords versus the squeezing strength. We set a target limit of $L_{\mathrm{st}} \sim 10^{-3}$.

.

where the elements X_{ij} and W_{ij} of the matrix X and W can be written as

$$X_{ij} = \sum_{l} x_{ri} x_{rj}^{*}, \quad W_{ij} = \mathrm{Tr}\left[\mathcal{N}\left(\hat{B}_i \otimes \hat{B}_j^{\dagger}\right)\right], \tag{S30}$$

respectively. From the solution to the above convex optimization problem we acquire the corresponding optimal recovery channel, where the optimal recovery operators can be given by a singular value decomposition of X_{opt},

$$\begin{aligned} X_{\mathrm{opt}} &= V\Omega V^{\dagger}, \\ R_{\mathrm{opt}} &= \sqrt{\Omega_r}\sum_i V_{ir} B_i, \end{aligned} \tag{S31}$$

where Ω_r is the singular value and V is an unitary matrix. We utilize Cvxpy for semidefinite convex optimization in Python [9, 10]. This recovery channel represents the optimal recovery channel, which defines the upper bound for QEC. As an example, we calculate the upper fidelity bounds for the optimal and conventional GKP codes at the time scales $\kappa\tau = 0.0004$ and $\kappa_\phi\tau = 0.0004/1.5$, as shown in Tab. S1. Our results show that, although both the optimal and conventional codes exhibit some noise bias for different encoded states, due to the incomplete satisfaction of the KL condition, the optimal code surpasses the conventional code for all six states.

TABLE S1. Comparison of the fidelity upper bounds for the optimal and conventional GKP codes

Encoding state	$\lvert 0_{\mathrm{L}}\rangle$	$\lvert 1_{\mathrm{L}}\rangle$	$(\lvert 0_{\mathrm{L}}\rangle + \lvert 1_{\mathrm{L}}\rangle)/\sqrt{2}$	$(\lvert 0_{\mathrm{L}}\rangle - \lvert 1_{\mathrm{L}}\rangle)/\sqrt{2}$	$(\lvert 0_{\mathrm{L}}\rangle + i\lvert 1_{\mathrm{L}}\rangle)/\sqrt{2}$	$(\lvert 0_{\mathrm{L}}\rangle - i\lvert 1_{\mathrm{L}}\rangle)/\sqrt{2}$
Optimal GKP	0.999968	0.999978	0.999964	0.999957	0.999954	0.999954
Conventional GKP	0.999918	0.999926	0.9999251	0.999919	0.999919	0.999919

IV. UNIVERSALITY OF THE NEURAL-NETWORK-BASED GKP CODEWORDS

Here, we investigate the universality of the neural network-based GKP codewords, including the performance of these approximate GKP codewords under various squeezing levels, the impact of small perturbations in the codeword coefficients, and the performance when the codeword coefficients are restricted to real numbers.

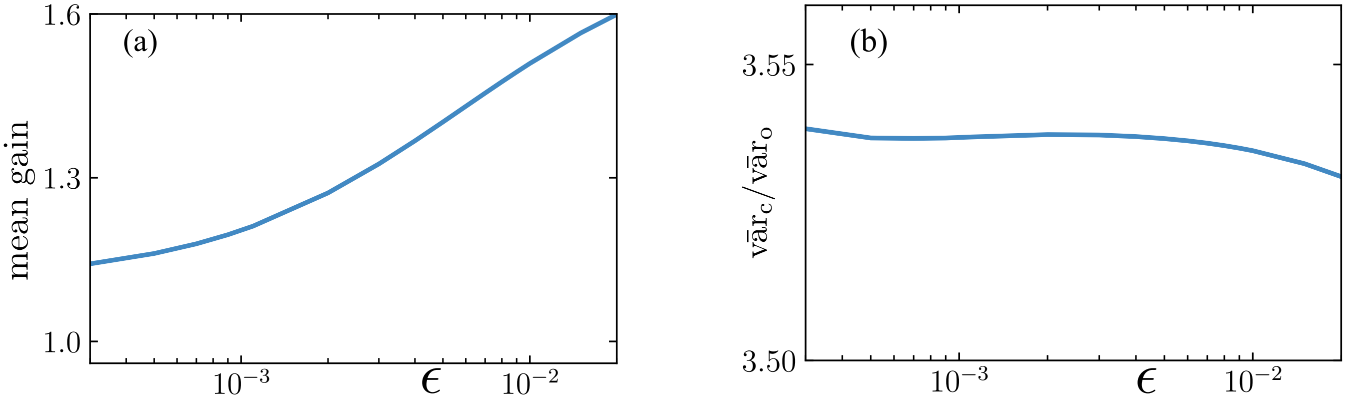

FIG. S3. (a) Mean gain of optimal GKP codewords versus conventional GKP codewords as a function of the noise magnitude of the coefficients $c_k^{(u)}$. (b) Mean-variance ratio of optimal and conventional approximate GKP codewords versus noise size ϵ.

A. Universality for various squeezing strengths

To illustrate the university of our proposal, we investigate the minimum number of squeezed coherent states required to optimize GKP under different squeezing levels. Our proposal effectively reduces the number of squeezed coherent states and enhances error correction performance under different strength squeezing levels, as shown in Fig. S2(a). Specifically, the optimized approximate GKP codewords lower the number of squeezed states to *seven* in each logical base (1/3 of the conventional approximate GKP for squeezing amplitudes $r > 1$) while maintaining better error correction performance.

The ideal stabilizers only offer rough stabilization for the approximate code at finite squeezing degrees. To address this, we propose more precise stabilizer operators, inspired by the special conventional GKP code in the limit $\zeta^{-1} = e^r \gg 1$,

$$\begin{aligned}\hat{S}_{q,\mathrm{ap}} &= \exp\left[i2\sqrt{\pi}\left(f_{11}\hat{q} + f_{12}\hat{p}\right)\right],\\ \hat{S}_{p,\mathrm{ap}} &= \exp\left[-i2\sqrt{\pi}\left(f_{21}\hat{q} + f_{22}\hat{p}\right)\right],\end{aligned}\tag{S32}$$

where the elements of the coefficient matrix $\mathbf{f} = [f_{11},\ f_{12};\ f_{21},\ f_{22}]$ can be complex numbers. The matrix elements have the relation $f_{11}f_{22} - f_{12}f_{21} = 1$, which preserves the commutation $\hat{S}_{q,\mathrm{ap}}\hat{S}_{p,\mathrm{ap}} = \hat{S}_{p,\mathrm{ap}}\hat{S}_{q,\mathrm{ap}}$.

We assume that the optimized stabilizers also follow this general form, with $|u_L\rangle = \hat{S}_{p,\mathrm{ap}}|u_L\rangle$ and $|u_L\rangle = \hat{S}_{q,\mathrm{ap}}|u_L\rangle$. Satisfying this condition is equivalent to satisfying the relations $\langle u_L|\hat{S}_{q,\mathrm{ap}}|u_L\rangle \approx 1$, $\langle u_L|\hat{S}_{p,\mathrm{ap}}|u_L\rangle \approx 1$, $\langle u_L|\hat{S}^\dagger_{q,\mathrm{ap}}\hat{S}_{q,\mathrm{ap}}|u_L\rangle \approx 1$, and $\langle u_L|\hat{S}^\dagger_{p,\mathrm{ap}}\hat{S}_{p,\mathrm{ap}}|u_L\rangle \approx 1$. To quantify how well the codewords satisfy these conditions, we define the following cost function,

$$L_{\mathrm{st}} = \sum_{u=0,1}\sum_{\hat{O}} |1 - \langle u_L|\hat{O}|u_L\rangle|^2,\tag{S33}$$

where $\hat{O} \in \left\{\hat{S}_{q,\mathrm{ap}},\ \hat{S}_{p,\mathrm{ap}},\ \hat{S}^\dagger_{q,\mathrm{ap}}\hat{S}_{q,\mathrm{ap}},\ \hat{S}^\dagger_{p,\mathrm{ap}}\hat{S}_{p,\mathrm{ap}}\right\}$. Therefore, we incorporate the above cost function into the total loss function. It is important to note that the Pauli operators in the logical code space are represented as $\hat{\sigma}_z = \hat{S}^{1/2}_{q,\mathrm{ap}}$ and $\hat{\sigma}_x = \hat{S}^{1/2}_{p,\mathrm{ap}}$, which can converge autonomously without the need to consider an additional loss function.

The coefficient matrix f for ideal GKP codewords has the elements $f_{11} = f_{22} = 1$ and $f_{12} = f_{21} = 0$. We define f_{er} as the distance between the stabilizer operators of the approximate and the ideal GKP codewords

$$f_{\mathrm{er}} = |f_{11} - 1| + |f_{22} - 1| + |f_{12}| + |f_{21}|\,.\tag{S34}$$

In Fig. S2(b), we simulate f_{er} as a function of the squeezing strength. As the squeezing increases, f_{er} decreases monotonically, showing that these stabilizer operators become closer to the stabilizer operators of the ideal codewords. Therefore, the optimal GKP codewords are valid and useful at different squeezing levels.

B. Robustness under disturbances

We investigate how profile perturbations of the coefficient $c_k^{(u)}$ impact the error correction performance of the encoded quantum states. The coefficients of the logical basis can be expressed as the mean values with small fluctuations due to imperfect control

$$c^{(u)} = c^{(u)} + \epsilon c^{(u)} \mathit{\Xi}, \tag{S35}$$

where ϵ is the magnitude of the noise with a value in $\epsilon \in (0, 0.02]$, and $\mathit{\Xi}$ is a random matrix whose elements are sampled from a uniform distribution over the interval $\Xi \in [-0.5, 0.5]$. Since the mean infidelity is proportional to $\bar{L}_{\rm er}$, we can define the average gain

$$\bar{G} = \bar{L}_{\rm er}({\rm ogkp})/\bar{L}_{\rm er}({\rm cgkp}), \tag{S36}$$

of the optimal GKP codewords compared to the best conventional ones. We show the average gain and the ratio $\bar{\rm var}_c(\bar{L}_{\rm er})/\bar{\rm var}_o(\bar{L}_{\rm er})$ for randomly generated noise $\mathit{\Xi}$ versus the noise magnitude ϵ in the Fig. S3 (a) and (b). The mean gains are always above one and increase with increasing noise magnitude ϵ; the variance of the loss function for the conventional GKP codewords remains more than 3.5 times that of the optimized GKP one. Our encoding demonstrates significantly greater robustness, whereas the conventional GKP codes are highly susceptible to noise, with the imperfect state preparation severely impairing error correction. This advantage arises from our codes avoiding 2/3 of the large-amplitude squeezed coherent states that are critical for the conventional codes, but challenging to prepare accurately. These states, particularly in optical systems, are regarded as a fundamental obstacle to the GKP state preparation. Thus, our codes enhance the stability and simplifies the state preparation, substantially reducing the impact of imperfections. This demonstrate that the noise resilience of the *optimized* GKP encoding outperforms that of the *conventional* GKP encoding, allowing for effective error correction and reducing the demand for extremely precise control to generate the GKP codewords.

C. The real coefficient GKP codewords

Here we restrict to the real coefficients $c_k^{(u)}$ to investigate the neural-network based GKP codewords. Our results show that the optimal GKP codewords at squeezing amplitude $r = 1.1$ still exhibit better quantum error correction ability than the best conventional one, where the number of squeezing states is *seven* and corresponding coefficients are

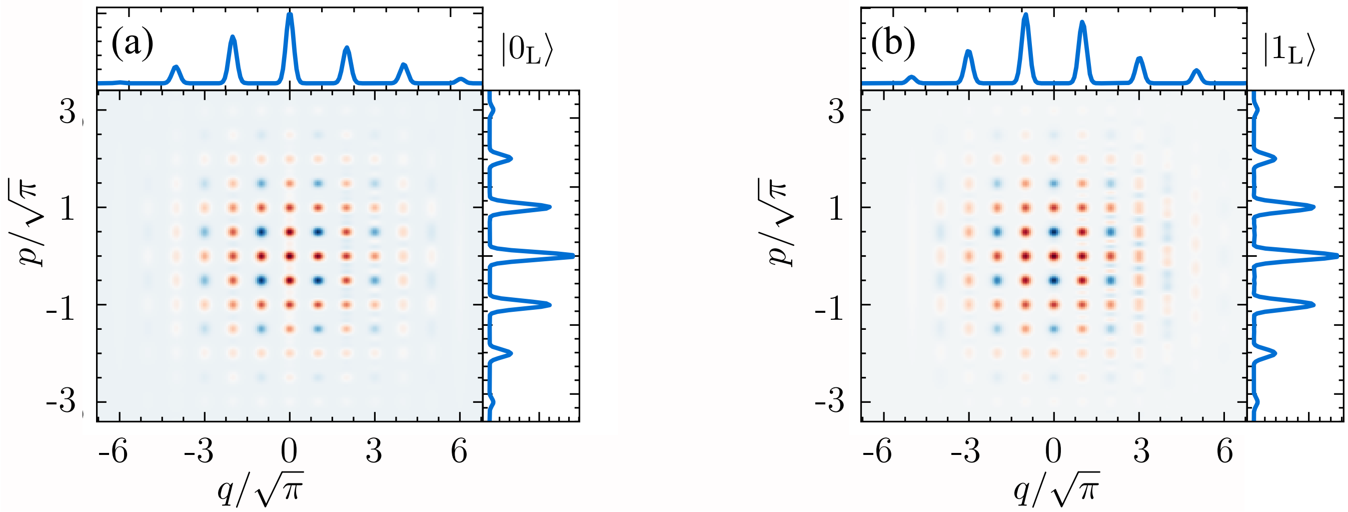

FIG. S4. Panels (a) and (b) give the Wigner functions for the optimal codewords $|0_L\rangle$ and $|1_L\rangle$ with real coefficients $c_k^{(u)}$ for $M = 3$, respectively.

$$\begin{aligned} c^{(0)}/\mathcal{N}(0) &= [0.054826,\ 0.228328,\ 0.381576,\ 0.470909,\ 0.334463,\ 0.243658,\ 0.118351], \\ c^{(1)}/\mathcal{N}(1) &= [0.114688,\ 0.258726,\ 0.375942,\ 0.354715,\ 0.229235,\ 0.163002,\ -0.039539]. \end{aligned} \tag{S37}$$

For these optimal codewords, the stabilizer operators match those in Eq. (S32), with the coefficient matrix

$$f=\begin{bmatrix} 0.999531+0.000695i & -0.000088+0.110767i \\ -0.000067-0.032467i & 1.004067-0.000703i \end{bmatrix} \tag{S38}$$

having an approximation level close to $L_{\mathrm{st}}\sim 10^{-3}$. Note that the error correction performance of this GKP state is only slightly poorer than that of the GKP state with complex coefficients offered in the main text. Moreover, we show the Wigner function of codewords with the real coefficients Eq. (S37) in Figs. S4 (a) and (b). Clearly, the projection of the Wigner function in the momentum coordinate system is consistent with that of the complex field. The main difference is the projection distribution in the position coordinates. This shows that optimizing GKP codewords for real coefficients is also a reliable and near-optimal alternative.

[1] D. Gottesman, A. Kitaev, and J. Preskill, Encoding a qubit in an oscillator, Phys. Rev. A **64**, 012310 (2001).

[2] H. P. Yuen, Two-photon coherent states of the radiation field, Phys. Rev. A **13**, 2226 (1976).

[3] S. M. Girvin, Introduction to quantum error correction and fault tolerance, SciPost Phys. Lect. Notes , 70 (2023).

[4] E. Knill, R. Laflamme, and L. Viola, Theory of quantum error correction for general noise, Phys. Rev. Lett. **84**, 2525 (2000).

[5] K. Audenaert and B. De Moor, Optimizing completely positive maps using semidefinite programming, Phys. Rev. A **65**, 030302 (2002).

[6] A. S. Fletcher, P. W. Shor, and M. Z. Win, Optimum quantum error recovery using semidefinite programming, Phys. Rev. A **75**, 012338 (2007).

[7] R. L. Kosut and D. A. Lidar, Quantum error correction via convex optimization, Quantum Inf. Process. **8**, 443 (2009).

[8] M. Reimpell and R. F. Werner, Iterative optimization of quantum error correcting codes, Phys. Rev. Lett. **94**, 080501 (2005).

[9] S. Diamond and S. Boyd, CVXPY: A Python-embedded modeling language for convex optimization, J. Mach. Learn. Res. **17**, 1 (2016).

[10] A. Agrawal, R. Verschueren, S. Diamond, and S. Boyd, A rewriting system for convex optimization problems, Journal of Control and Decision **5**, 42 (2018).