Coherent optical communications enhanced by machine intelligence

MACHINE
LEARNING
Science and Technology

PAPER

CrossMark

Coherent optical communications enhanced by machine intelligence

Sanjaya Lohani and Ryan T Glasser

Tulane University, New Orleans, LA 70118, United States of America
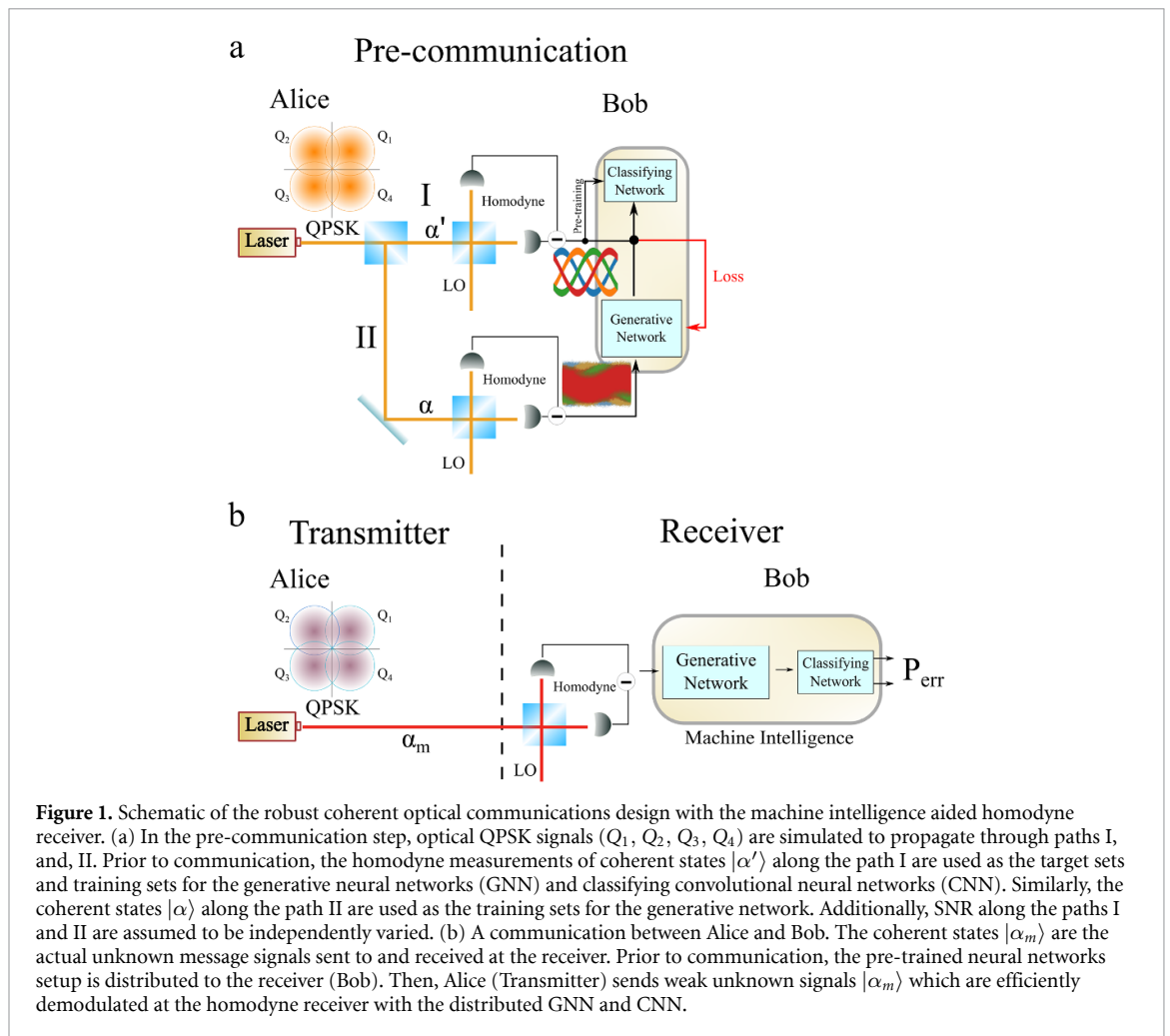
E-mail: slohani@tulane.edu and rglasser@tulane.edu
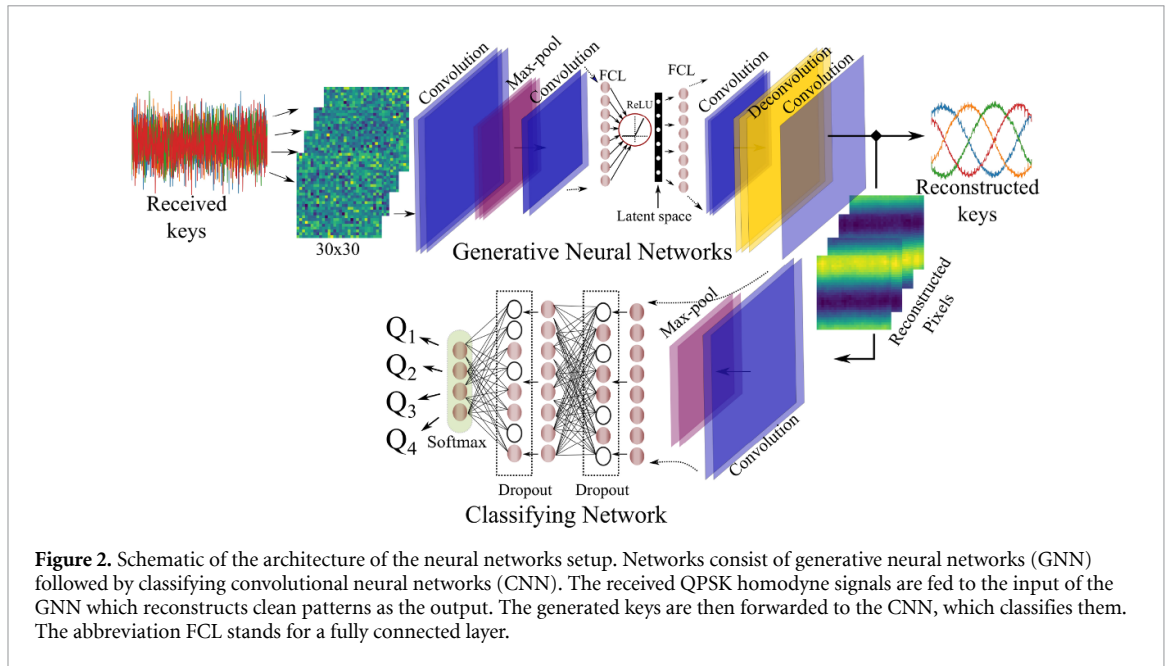
Abstract

Accuracy in discriminating between different received coherent signals is integral to the operation of many free-space communications protocols, and is often difficult when the receiver measures a weak signal. Here we design an optical communication scheme that uses balanced homodyne detection in combination with an unsupervised generative machine learning and convolutional neural network (CNN) system, and demonstrate its efficacy in a realistic simulated coherent quadrature phase shift keyed (QPSK) communications system. Additionally, we design the neural network system such that it autonomously learns to correct for the noise associated with a weak QPSK signal, which is distributed to the receiver prior to the implementation of the communications. We find that the scheme significantly reduces the overall error probability of the communications system, achieving the classical optimal limit. We anticipate that these results will allow for a significant enhancement of current classical and quantum coherent optical communications technologies.

1. Introduction

Nonorthogonality of coherent states allows for their use in various quantum and classical optical technology systems, such as quantum networks (Loock 2011) and quantum metrology (Wiseman & Milburn 2009, Armen *et al* 2002), as well as classical optical communications schemes (Grosshans *et al* 2003, Betti *et al* 1995, Grosshans & Grangier 2002). Coherent states are relatively tolerant to losses and nonlinearities in communications channels (Giovannetti *et al* 2004). Additionally, with the introduction of application-specific integrated circuits (ASICS) and field-programmable gate arrays (FPGAs), research interest involving coherent optical communications is rapidly growing (Yoshida *et al* 2011). Furthermore, coherent optical phase shift keying, in particular QPSK, communications systems have been demonstrated to have high data transfer rates and low complexity (Patnaik & Sahu 2013). Due to such benefits, coherent optical QPSK is extensively employed in terrestrial (Feng *et al* 2018) as well as inter-satellite communications (Bindushree *et al* 2014), such as the global positioning system (GPS), and common data links (CDL). However, the accurate discrimination of coherent states is always a barrier for efficient classical communications, in particular where the receiver site detects weak, low signal-to-noise ratio (SNR) coherent states such as in deep space communications (Kaushal & Kaddoum 2017). Realistic coherent optical communications systems involve the propagation of such signals through various turbulent media that may decrease the SNR (Ke *et al* 2018), thus increasing the difficulty in discriminating between the four possible phase-shifted keys. As a result, the noisy received QPSK signals can significantly limit the ability to establish such communications in a real-world environment. Here we make use of generative machine learning techniques (Vincent *et al* 2008) in combination with a convolutional neural network (Lohani *et al* 2018) to design a communications system that is robust to signal fading (or weak coherent signals) and demonstrate its ability to significantly reduce the error probability in discriminating between received coherent QPSK signals in a realistic simulated communications setting.

Figure 1. Schematic of the robust coherent optical communications design with the machine intelligence aided homodyne receiver. (a) In the pre-communication step, optical QPSK signals (Q_1, Q_2, Q_3, Q_4) are simulated to propagate through paths I, and, II. Prior to communication, the homodyne measurements of coherent states $|\alpha'\rangle$ along the path I are used as the target sets and training sets for the generative neural networks (GNN) and classifying convolutional neural networks (CNN). Similarly, the coherent states $|\alpha\rangle$ along the path II are used as the training sets for the generative network. Additionally, SNR along the paths I and II are assumed to be independently varied. (b) A communication between Alice and Bob. The coherent states $|\alpha_m\rangle$ are the actual unknown message signals sent to and received at the receiver. Prior to communication, the pre-trained neural networks setup is distributed to the receiver (Bob). Then, Alice (Transmitter) sends weak unknown signals $|\alpha_m\rangle$ which are efficiently demodulated at the homodyne receiver with the distributed GNN and CNN.

Generative machine intelligence systems have recently been applied to a variety of research areas (Sanchez-Lengeling & Aspuru-Guzik 2018, Jang *et al* 2018, Li *et al* 2018, Donahue *et al* 2018, Torlai *et al* 2018). Here we implement an unsupervised denoising autoencoder (Vincent *et al* 2008) as the generative neural network (GNN), a concept which has been demonstrated to be useful in multiple applications (Gondara 2016, Fichou & Morlock 2018, Cheng *et al* 2018). Recently, several research groups have shown the power of machine intelligence in the context of coherent optical communications (Wang *et al* 2017, Zhang *et al* 2018, Lohani & Glasser 2018, Khan *et al* 2017, Fan & Wu 2017, Kulin *et al* 2017). Here we use a generative neural network, in combination with a convolutional neural network (CNN), to make clean reconstructions and accurate classifications of received optical QPSK signals even for extremely low SNR scenarios, which greatly reduces the overall error probability of the communications design, achieving the classical optimal (i.e homodyne limit (Proakis & Salehi 2001)) or standard quantum limit (SQL). In previous optical communications techniques using machine learning, unknown received optical signals (including images) have been classified with high accuracy using only a CNN as the classifier. It is, however, desirable for schemes with only a CNN as the classifier to have a training set with a large number of known homodyne measurements, which is always unbounded. This limits the discrimination efficiency of the communications design with respect to random signal (amplitude) fading of coherent states. Additionally, such setups with only a CNN as a classifier are based on supervised learning that requires a labelling for each training homodyne measurement. In practice, it is hard to accurately label the homodyne outputs of a received QPSK signal, particularly for very low SNRs, which may require an extra detector that further adds some discrimination error increasing the overall error probability of the communication design. The present unsupervised learning strategy does not require any labelling for the weak (low SNR) coherent states, which are autonomously fed to the networks and reconstructed at the GNN outputs. Afterwards, the generated keys are classified by a CNN purely trained with the homodyne data associated with high SNR QPSK signals (desired), which are easily distinguished and labelled, prior to the start of communications (i.e. pre-trained before messages are sent and received). These novel aspects of the current design pave the way towards the

Figure 2. Schematic of the architecture of the neural networks setup. Networks consist of generative neural networks (GNN) followed by classifying convolutional neural networks (CNN). The received QPSK homodyne signals are fed to the input of the GNN which reconstructs clean patterns as the output. The generated keys are then forwarded to the CNN, which classifies them. The abbreviation FCL stands for a fully connected layer.

robust and realistic implementation of homodyne receivers in efficiently demodulating signals in the coherent free space optical communications regime.

The overall communications system design is shown in figure 1. A laser beam is split into two paths, path I and path II, with a beam splitter. The branches I and II are used in training the networks prior to communication. Here the branching ensures the realistic implementation of an unsupervised learning (autonomous learning) with the neural networks setup. Note that the QPSK signals $|\alpha'\rangle$, and $|\alpha\rangle$ on path I and II, respectively, share the same QPSK state (key value) so that the homodyne detectors on both paths always read the same key value regardless the strength of noise (SNR) in either branch as shown in figure 1(a). The QPSK signal $|\alpha_m\rangle$ ('m' for message) on the communications channel is assumed to be an unknown message key that has been sent and received at the receiver end as shown in figure 1(b). In the prior communication setup, the homodyne outputs of the QPSK signals $|\alpha\rangle$ on path II is fed to the input of the GNN, which reconstructs a new signal measurement at its output that is finally compared with the homodyne output of signals $|\alpha'\rangle$ from path I. Next we optimize the reconstruction loss and update the GNN parameter space (see 'Methods'). At the same time we use the homodyne measurements of the QPSK signals $|\alpha'\rangle$ to train the classifying CNN network (see 'Methods'). Note that the networks are pre-trained, and this pre-training may take place locally, then be distributed prior to implementation of the communications (which is the weak signal $|\alpha_m\rangle$ sent). Finally, the machine intelligence aided homodyne receiver classifies unknown QPSK message signals $|\alpha_m\rangle$ at the receiver and the corresponding error probability is evaluated. Examples of the homodyne measurement of a received QPSK signal and corresponding clean reconstruction are shown in figure 2. Furthermore, we evaluate the error probability for various combinations of $|\alpha\rangle$ (or $|\alpha_m\rangle$) and $|\alpha'\rangle$, various scanning phase ranges of the local oscillator, various SNR values of the transmitted message signal $|\alpha_m\rangle$, and wide range of phase noise. Finally, we show a significant improvement in the overall error probability when the neural network system is used.

2. Methods

2.1. Architecture of the GNN

The GNN consists of three sections—an encoder, latent space, and a decoder (generator). The encoder begins with a two dimensional convolutional layer with a kernel of size [5,5], stride length of 1, batch size of 10, feature mappings of 20, same padding, and a ReLU activation function. After this we apply a dropout with a rate of 20% followed by a max-pool layer with a kernel of size [2,2] with a stride length of 2. Then, again, we connect a two dimensional convolutional layer to the max-pool with the same parameters and dropout as discussed above. Next we attach a fully connected layer with the ReLU activation function, where a number of neurons is always equal to the size of the latent space. Note that for the consistency we always choose the size of the latent space as half of the input dimension (Lohani & Glasser 2019). For example, input dimensions of 30×30, and 16×16, respectively, have a 15×15, and 8×8 sized latent space. Finally, the encoder convolutes the input dimension of [10 (batch size), width, height, 1 (input channel)] to [10, latent

space dimension]. At the beginning of the decoder, the latent space is connected to a fully connected layer with [width/2× height/2× 20 (feature mappings)] neurons, which is followed by a dropout with a rate of 20%. After this we apply a convolutional layer with stride length of 1 and transpose-convolutional (or deconvolutional) layer with stride length of 2, each with a kernel of size [5,5], feature mappings of 20, same padding, and a ReLU activation function. Note that each layer is followed by a dropout with a rate of 20%. Finally a convolutional unit with a single feature mapping generates a clean, or less noisy, homodyne measurement as the output. Here, the generator decodes the latent space of size [10 (batch size), latent space dimension] into the outputs of size [10 (batch size), width, height]. In order to optimize the clean reconstructions, a square reconstruction loss (error) $\mathbf{L}\big(GNN(\alpha), GNN(\alpha')\big)$ is evaluated, where $GNN(\alpha)$ is the output of the GNN, and $GNN(\alpha')$ is the target. Finally, we minimize the average reconstruction loss given by equation (1) using adamoptimizer of tensorflow (Abadi *et al* 2015) with a learning rate of 0.008.

$$\theta, \theta' = \theta, \theta' \operatorname{argmin} \frac{1}{N} \sum_i^N \mathbf{L}\big(GNN(\alpha), GNN(\alpha')\big), \tag{1}$$

where θ and θ' represent the encoder and decoder parameter space, respectively. The schematic of the GNN is shown in figure 2.

2.2. Architecture of the classifying CNN

The classifying CNN network begins with a convolutional unit with a kernel of size [2,2], batch size of 1, feature mappings of 10, same padding, stride length of 1, and a ReLU activation function. This is followed a max-pool layer with a kernel of size [2,2] and stride length of 2. Then we connect fully connected layers (FCLs) with 400 neurons and 50 neurons, which are consecutively followed by dropout layers with a rate of 80%, and 40%, respectively. Note that we use ReLU activations for both FCLs. Next we attach a final FCL with 4 neurons with a linear activation function to the end, followed by a softmax operation. In order to train a CNN to classify the generated clean homodyne outputs, we always use the target set of the GNN i.e. the homodyne measurements at the signal strength of $|\alpha'|$ as shown in path I of figure 1. We randomly simulate 200 homodyne outputs per each QPSK key for the given $|\alpha'|$ as discussed in the results section, resulting in a total of 800 homodyne outputs. The data set is split into a training set with 170 measurements and a test set with 30 measurements, again, per each QPSK. The target of each QPSK is set as the one-hot vector output, for example, the first target key is [1,0,0,0], the second key is [0,1,0,0], and so on. After this the parameter space of the CNN is optimized by minimizing a softmax cross-entropy loss function using adamoptimizer with a learning rate of 0.001 up to 10 epochs. The neural networks' hyper-parameters are manually optimized as discussed in (Lohani *et al* 2018). Note that pre-trained CNN network has unity accuracy with respect to the test homodyne measurements. The schematic of the classifying CNN is shown in figure 2.

Results

Here we simulate QPSK coherent signals $|\alpha'^k\rangle$, $|\alpha^k\rangle$ and $|\alpha_m^k\rangle$ for k ∈ {1,2,3,4} with,

$$\alpha'^k = |\alpha'|e^{i\phi^k}, \alpha^k = |\alpha|e^{i\phi^k}, \quad \text{and} \quad \alpha_m^k = |\alpha_m|e^{i\phi^k},$$
$$\text{such that} \quad \phi^k = \left(k - \frac{1}{2}\right)\frac{\pi}{2}, \tag{2}$$

where each coherent state has some mean amplitude (e.g. $|\alpha|$) and phase (ϕ). In order to simulate a balanced (one output from a 50:50 beam splitter is subtracted from the other as shown in figure 1) homodyne measurement, we use a local oscillator, $\beta = |\beta|e^{i\gamma}$, with amplitude $|\beta| >> |\alpha|$. Note that we set $\beta = 100$ for all of the simulations presented in this paper. As a result, the mean signal, $\langle n \rangle$, and variance, $(\Delta n)^2$, at the detector are given by

$$\langle n \rangle = 2|\beta|\langle \alpha|X_\gamma|\alpha\rangle \quad \text{and} \quad (\Delta n)^2 = |\beta|^2,$$
$$\text{given that} \quad X_\gamma = \frac{a^\dagger e^{i\gamma} + a e^{-i\gamma}}{2}, \quad \text{and} \quad \gamma : \gamma + \pi/2 \tag{3}$$

where $a^\dagger$ and a are the raising and lowering operators, respectively. Since the minimum error probability (P_{err}) of discriminating between the QPSK signals (keys) are bounded below classically by the homodyne limit(Proakis & Salehi 2001), and quantum mechanically by the Helstrom limit(Helstrom 1969), we design and demonstrate that the present communication setups achieve the classical minimum error limit even for extremely low SNR scenarios. In order to calculate the SQL (P_{err}^{HD}) and the Helstrom limit (P_{err}^{Hel}), we use the

following relations, which are discussed in detail in (Proakis & Salehi 2001, Becerra *et al* 2013, Izumi *et al* 2012),

$$P_{err}^{HD} = \mathbf{erfc}(\frac{\alpha}{\sqrt{2}})\Big[1 - \frac{1}{4}\mathbf{erfc}(\frac{\alpha}{\sqrt{2}})\Big];$$

$$P_{err}^{Hel} = 1 - \frac{1}{8}e^{-|\alpha|^2}\Big(\sqrt{\cosh|\alpha|^2 + \cos|\alpha|^2}$$

$$+ \sqrt{\sinh|\alpha|^2 + \sin|\alpha|^2} + \sqrt{\cosh|\alpha|^2 - \cos|\alpha|^2} + \sqrt{\sinh|\alpha|^2 - \sin|\alpha|^2}\Big)^2, \tag{4}$$

where $\mathbf{erfc}(u) = \frac{2}{\sqrt{\pi}}\int_u^\infty e^{-u^2}\,du$, is the complementary of the error function.

Next, we implement a GNN consisting of three sections—an encoder, latent space, and a generator. The encoder learns the important features of the signals and stores them into a smaller dimension, which is a latent space. The latent space is then decoded through the generator which finally reconstructs the desired clean QPSK homodyne outputs. The implemented GNN uses convolutional units as the encoder and transpose-convolutional units as a decoder. The encoder contains two convolutional layers, a single max-pool layer, and a single fully connected layer (FCL) to the latent space, whereas the generator starts with a FCL followed by a convolutional layer, a transpose-convolutional layer, and again a convolutional layer as shown in figure 2. Similarly, in order to classify the reconstructed, as well as uncorrected (noisy) QPSK homodyne outputs, we use a CNN at the end. The CNN consists of a single convolutional layer followed by max-pooling layer, and two FCLs. Finally, the FCL is connected with an output layer where classification decisions are made as shown in figure 2. The architectures of the GNN and CNN are described in detail in the 'Methods' section. The error probability introduced by the network ($P_{err}^{network}$) is further associated with the error probability while labeling the homodyne outputs for the training sets of the CNN classifier (P_{err}^{label}), error probability of the pre-trained CNN (P_{err}^{tCNN}), and error probability for making predictions of the unknown message signal ($P_{err}^{\alpha_m}$), which is equal to the ratio of the total number of incorrect classifications to the total number of optical QPSK signal received, resulting in the overall error probability (P_{err}) in discriminating the received noisy QPSK signals at the receiver given by,

$$P_{err} = 1 - (1 - P_{err}^{HD})(1 - P_{err}^{network})$$
$$= 1 - (1 - P_{err}^{HD})(1 - P_{err}^{label})(1 - P_{err}^{tCNN})(1 - P_{err}^{\alpha_m}). \tag{5}$$

With the aid of the unsupervised GNN in the communication design, we are able to train (pre-train) the CNN classifier such that both P_{err}^{label} and P_{err}^{tCNN} are always 0, which is practically possible with very high SNR data set as discussed in the next paragraph. This leads the overall error probability of the communication design to equation (6),
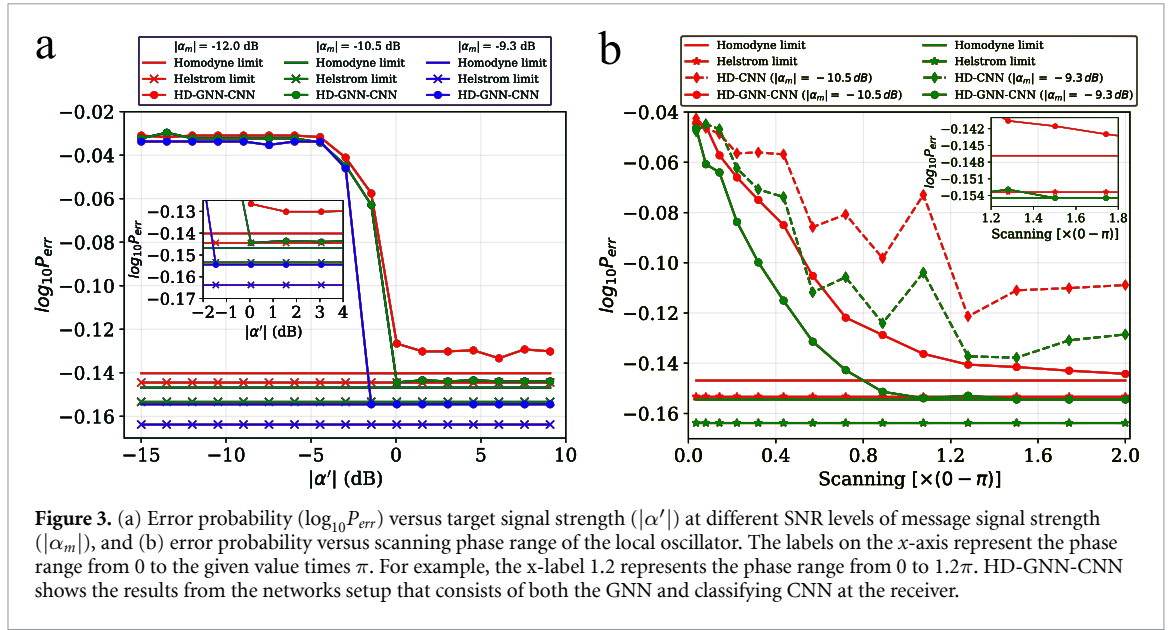
$$P_{err} = 1 - (1 - P_{err}^{HD})(1 - P_{err}^{\alpha_m}). \tag{6}$$

We then evaluate the error probability and plot it as a $\log_{10}P_{err}$ and a signal strength $|\alpha|$ in decibels, $|\alpha|\,dB = 10\log_{10}(|\alpha|)$.

First we evaluate the P_{err} with respect to the amplitude $|\alpha'|$ along the path I, given different SNRs in path II. Here we scan the local oscillator phase from 0 to 2π with a grid consisting of 900 points. We randomly simulate 200 homodyne outputs for each QPSK signal for 17 different values of $|\alpha'|$ from -15 dB to 9.08 dB, and set them as the target patterns at the output of the GNN. Similarly, we do the same for path II and randomly simulate 200 homodyne outputs for each QPSK signal for $|\alpha| = -12.0$ dB, -10.5 dB, and -9.3 dB, and feed them to the encoder of GNN. Note that before feeding the signals to the GNN, we convert the homodyne outputs (from path I and II) into corresponding 30×30 pixels images as shown in figure 2. In order to optimize $|\alpha'|$ at various values of $|\alpha|$, we keep $|\alpha|$ fixed at -12.0 dB, -10.5 dB, and -9.3 dB, and vary $|\alpha'|$ for all of them separately, such that 17 different GNN configurations are trained up to 150 epochs (approximately 8 minutes) separately for each $|\alpha|$, for a total of 51 pre-trained GNNs. Also, we use the same $|\alpha'|$ data set, separately, to pre-train the classifying CNN networks as shown in figure 1(a). Then the pre-trained networks are distributed to the receiver site (Bob).
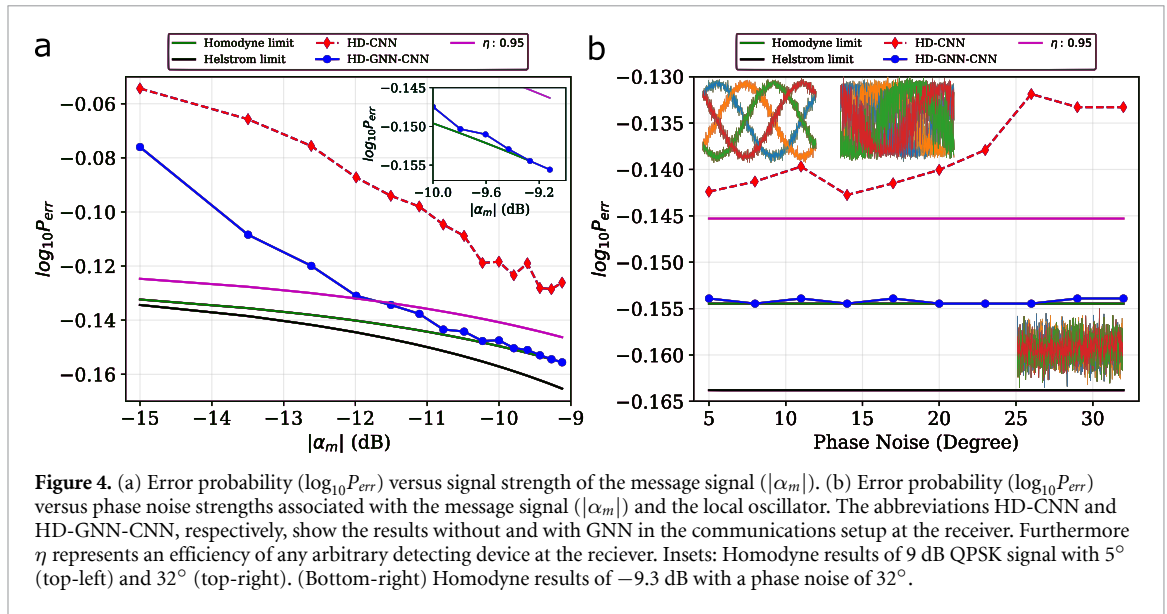
In order to establish a communication setup as shown in figure 2(b), we randomly simulate 360 homodyne outputs (90 per each QPSK key) for each $|\alpha_m| = -12.0$ dB, -10.5 dB, and -9.3 dB as the test sets. Note that message data and training data for the pre-communication setup share no similarity as they are randomly simulated at different times. Finally at the receiver, the homodyne measurements of $|\alpha_m|$ are fed to the distributed pre-trained GNN which generates desired homodyne measurement as the output. After that, in order to calculate P_{err}, the reconstructed images (generated homodyne measurement outputs) from the GNN are forwarded to a pre-trained CNN and the corresponding P_{err} is measured, the results of which

Figure 3. (a) Error probability ($\log_{10} P_{err}$) versus target signal strength ($|\alpha'|$) at different SNR levels of message signal strength ($|\alpha_m|$), and (b) error probability versus scanning phase range of the local oscillator. The labels on the *x*-axis represent the phase range from 0 to the given value times π. For example, the x-label 1.2 represents the phase range from 0 to 1.2π. HD-GNN-CNN shows the results from the networks setup that consists of both the GNN and classifying CNN at the receiver.

are shown in figure 3(a). Note that the distributed pre-trained networks reconstruct and then classify $|\alpha_m|$ on the order of milliseconds. The P_{err} for $|\alpha_m| = -12.0$ dB, -10.5 dB and -9.3 dB at various $|\alpha'|$ are shown by the solid red, green, and blue curves, respectively, with the same color dotted lines representing the corresponding homodyne limit (P_{err}^{HD}), and black solid line representing the Helstrom limit (P_{err}^{Hel}). We find an improvement in P_{err} begins when $|\alpha'| \geq -4.5$ dB for all $|\alpha_m| = -12.0$ dB, -10.5 dB, and -9.3 dB. As expected we obtain better reconstructions and less P_{err} as we increase $|\alpha'|$ up to -1.5 dB for $|\alpha_m| = -12.0$ dB, and around 0 dB for, -10.5 dB, and -9.3 dB, after which they begin to saturate. Additionally, we show that the system's P_{err} achieves the corresponding homodyne limit for $|\alpha_m| = -9.3$ dB at $|\alpha'| = -1.5$ dB. Similarly we find a difference of 2.4×10^{-3}, and 6.8×10^{-3}, respectively, between the P_{err} and corresponding homodyne limit for $|\alpha_m| = -10.5$ dB, and -12.5 dB when $|\alpha'| = 0.05$ dB, and 6.1 dB. Moreover, the corresponding P_{err} for $|\alpha'|$ from -2 dB to 4 dB at the given $|\alpha_m|$ are zoomed in and shown in the inset of figure 3(a). After taking into account the various SNR scenarios, we set $|\alpha'|$ relatively high at 9 dB as the optimized value for all simulations and results discussed in the following paragraphs. Note that this signal is used in pre-training the neural networks prior to communication, and is not the signal that would be sent through a realistic communications channel (which is $|\alpha_m|$, and is simulated to be much weaker as discussed in the following).

Next, we investigate the P_{err} as the phase scanning range of the detection system is varied. We use the same training-test data and network settings as discussed above. In order to simulate training and test sets at various phase scanning ranges, we slice the grid of the homodyne outputs into a number of segments, for example as discussed in previous paragraph, into a grid of 30×30 (i.e. $0 - 900$ points) which represents a phase scanning from 0 to 2π. Here, we slice it into 28×28 ($0 - 784$ points), 26×26 ($0 - 676$ points), 24×24 ($0 - 576$ points) and so on up to 4×4 ($0 - 16$ points), for a total of 14 various scanning ranges. These correspond to phase scanning ranges from $0 - 1.74\pi$, $0 - 1.5\pi$, $0 - 1.28\pi$, and so on up to $0 - 0.04\pi$, respectively. Note that we separately train the GNN and CNN for each scanning range. Also we set the latent space dimension of the GNN to half of the input dimension for a given range. For example, the input of 28×28 and 12×12 points, respectively, have latent space sizes of 14×14 and 6×6 points. With the GNN pre-trained for an $|\alpha| = $ -10.5 dB, and -9.3 dB, and the CNN pre-trained with $|\alpha'| = 9$ dB, we simulate sending a message signal $|\alpha_m| = -10.5$ dB, and -9.3 dB (different from training sets) and calculate the P_{err} at the given scanning range. Here we reconstruct the 360 homodyne measurements (90 per each QPSK) for the given message set $|\alpha_m|$ and predict the corresponding QPSK value using the CNN, which is finally used to evaluate the P_{err}. The P_{err} resulting from the networks with and without the GNN versus scanning range is shown in figure 3(b). The improvement in P_{err} for $|\alpha_m| = -10.5$ dB is shown by red curves, where the dotted red curve, solid red curve, and horizontal red line represent the P_{err} without the GNN, with the GNN, and the homodyne limit, respectively. Similarly, the dotted green curve, solid green curve and a horizontal green line represent the P_{err} without the GNN, with the GNN, and the homodyne limit for $|\alpha_m| = -9.3$ dB, respectively. As discussed above, the black horizontal line is the Helstrom limit. We find a gradual improvement in the P_{err} as we increase the scanning phase range of the local oscillator. Furthermore, with the aid of the GNN in the network and a scanning range of $0 - 0.88\pi$, we show significant improvement in P_{err}

Figure 4. (a) Error probability ($\log_{10} P_{err}$) versus signal strength of the message signal ($|\alpha_m|$). (b) Error probability ($\log_{10} P_{err}$) versus phase noise strengths associated with the message signal ($|\alpha_m|$) and the local oscillator. The abbreviations HD-CNN and HD-GNN-CNN, respectively, show the results without and with GNN in the communications setup at the receiver. Furthermore η represents an efficiency of any arbitrary detecting device at the reciever. Insets: Homodyne results of 9 dB QPSK signal with 5° (top-left) and 32° (top-right). (Bottom-right) Homodyne results of −9.3 dB with a phase noise of 32°.

from −0.098 to −0.128, and −0.124 to −0.151 for $|\alpha_m| = -10.5$ dB, and −9.3 dB, respectively. Note that the homodyne limit for $|\alpha_m| = -10.5$ dB, and −9.3 dB are −0.147 and −0.154, respectively. Additionally, we find the P_{err} with the aid of the GNN is minimized and achieves the corresponding homodyne limit when the scanning range is greater than or equal to $0 - 1.14\pi$ for $|\alpha_m| = -9.3$ dB. A minimum difference of 2.6×10^{-3} between the GNN-aided P_{err} and homodyne limit is achieved for $|\alpha_m| = -10.5$ dB at a scanning range of $0 - 2\pi$. The P_{err} corresponding to a scanning range of $0 - 1.28\pi$, $0 - 1.74\pi$, and $0 - 2\pi$ are zoomed in and shown in upper right inset of figure 3(b).

We now turn to investigating the improvement of the P_{err} at various SNR levels of transmitted message signals $|\alpha_m|$. In order to generate training input sets for the GNN-network, we randomly simulate 200 homodyne measurements for each QPSK key for 15 values of $|\alpha|$ from −15.0 dB to −9.12 dB. As mentioned earlier, we set $|\alpha'|$ to 9 dB, and the corresponding 200 homodyne outputs for each QPSK signal are used as the target for the GNN. The same $|\alpha'|$ is used to train the CNN. Note that we train separately the GNN for each value of $|\alpha|$, giving a total of 15 GNNs, while the classifying network (CNN) remains the same (as they share same target of the GNN). Similarly, we randomly generate 90 homodyne measurements for each QPSK for 15 values of $|\alpha_m|$ from −15.0 dB to −9.12 dB as the message set separately. As a result, message sets and training sets again share no similarity. With the pre-trained GNN and CNN (distributed), we evaluate the P_{err} with and without the GNN in the networks, results of which are shown in figure 4(a). The dotted red and solid blue curves represent the P_{err} without the GNN and with the GNN in the networks, respectively. Similarly, a solid green and a black line represent the homodyne and Helstrom limits for various values of $|\alpha_m|$, where as a solid magenta line represents the P_{err} for any arbitrary classifier or device with an efficiency (η) of 95%. With the aid of the GNN in the networks, we achieve a remarkable improvement in P_{err} from −0.119 to −0.147, which nearly reaches the optimal homodyne limit of -0.148 for $|\alpha_m| = -10.23$ dB. Similarly, for $|\alpha_m| = -9.27$ dB, the P_{err} achieves an improvement from −0.129 to the optimal homodyne limit as shown in upper right inset of figure 4(a). Furthermore, even for a very low SNR of $|\alpha_m| = -13.49$ dB, we significantly reduce the difference between P_{err} and the homodyne limit from 6.9×10^{-2} to 2.7×10^{-2}. Additionally, for $|\alpha_m| \approx -12$ dB, we find that the P_{err} curve with the GNN in the networks crosses and beats the P_{err} curve with any 95% efficient detector at the receiver.

Finally, we significantly lower the P_{err} for the various strengths of phase noise associated with the transmitted message signals $|\alpha_m|$ and the local oscillator, which is shown in figure 4(b). In order to simulate the phase noise at the receiver, we introduce a phase offset of δ in the mean signal, which is $\langle n \rangle = 2|\alpha||\beta|\cos(\phi - \gamma + \delta)$ such that $\delta \in N(0, \rho)$, a normal distribution with mean 0 and variance ρ^2. In order to train the networks we keep the signal strength of $|\alpha'| = 9dB$ and $|\alpha| = -9.3dB$ fixed along the path I and II, respectively, as discussed earlier. As homodyne measurements along the path I are used as the target and training sets for the GNN and classifying CNN, respectively, we assume that this path has only phase noise strength (ρ) of $\pi/36$ radians (5°), and randomly simulate 200 homodyne outputs for each QPSK signal. At the same time, we simulate to vary ρ from $\pi/36$ to $8\pi/45$ radians (32°) along the path II, and again randomly generate 200 homodyne outputs for each QPSK signal and ρ, which are fed to the input of GNN. Examples of homodyne measurements of the optical QPSK signal along the path I with ρ of 5° (top-left) and

32° (top-right) are shown in the inset of figure 4(b). After the networks are pre-trained and distributed, we randomly simulate 90 homodyne outputs for each QPSK message signal with $|\alpha_m| = -9.3dB$ and given ρ. Note that the training homodyne sets of the networks and the message homodyne sets again share no similarity. An example for a message QPSK homodyne output with the ρ of 32° is shown in the bottom-right inset of figure 4(b). First, we train the networks separately for each ρ and evaluate the P_{err}, the results of which are shown in figure 4(b). The green, black and magenta horizontal lines, respectively, represent the Homodyne limit, Helstrom limit, and P_{err} introduced by an arbitrary $\eta = 95\%$ efficient detecting device. The blue solid and red dotted curves show the P_{err} with the GNN and without the GNN in the networks, respectively. With the aid of GNN, we find a significant reduction in the P_{err} for the wide range of phase noise strengths from 5° to 32°. For example with the addition of GNN, the P_{err} for the phase noise strength of 8°, 20°, and 26° are reduced to the Homodyne limit.

3. Discussion

In conclusion, we have designed a novel coherent optical communications setup that efficiently demodulates weak coherent optical QPSK states with a robust machine intelligence aided balanced homodyne receiver. The developed state-of-the-art GNN and CNN system, in combination, corrects for coherent QPSK signals associated with a wide range of SNRs, resulting in significantly reduced overall error probability of the communications system, either achieving or approaching the classical optimal limit. Additionally, by using the same network system, we also reduce the discrimination error probability for various scanning ranges of the local oscillator at different SNRs. In addition, we illustrate that the network design is robust for the wide range of phase noise associated with the optical signal and the local oscillator as well. Furthermore, with the aid of the unsupervised GNN, which reconstructs a clean and desired quadrature measurement at its output, we have minimized the error probability with a CNN that is exclusively trained with homodyne measurements associated with high SNR (desired) coherent QPSK signals. This allows bypassing the need for extremely large CNN training sets that would require various noises which are not only difficult to label, but are also unbounded. The present advances in QPSK demodulation and classification are essential to the robust performance of realistic coherent optical communications systems, and we anticipate that the presented techniques may directly be applied to an enhancement of current classical and quantum communications protocols (Bacco *et al* 2019, Eriksson *et al* 2019).

Author contributions statement

S L conceived and designed the neural network system, and ran all simulations. R T G supervised the project. Both authors prepared the manuscript.

Data availability

The data that support the findings of this study are available from the corresponding authors on reasonable request.

Competing interests

The authors declare no competing interests.

ORCID iD

Sanjaya Lohani ⬤ https://orcid.org/0000-0003-0699-0669

References

Abadi M *et al* 2015 TensorFlow: Large-scale machine learning on heterogeneous systems Software available from tensorflow.org https://www.tensorflow.org/

Armen M A, Au J K, Stockton J K, Doherty A C and Mabuchi H 2002 *Phys. Rev. Lett.* **89** 133602

Bacco D *et al* 2019 *Commun. Phys.* 2 1–8

Becerra F E, Fan J, Baumgartner G, Goldhar J, Kosloski J T and Migdall A 2013 *Nat. Photon.* **7** 147–52

Betti S, Marchis G D and Iannone E 1995 *Coherent Optical Communications Systems* 1st edn (New York: Wiley-Interscience)

Bindushree A P, Nataraju A, Vijesh T and Laxmiprasad A 2014

Cheng X, Zhang L and Zheng Y 2018 *Comput. Method. Biomec.* 6 248–52

Donahue C, Li B and Prabhavalkar R 2018 *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)* IEEE pp 5024–8

Eriksson T A *et al* 2019 *Commun. Phys.* 2 1–8

Fan M and Wu L 2017 *2017 Int. Conf. on Communication, Control, Computing and Electronics Engineering (ICCCCEE)* pp 1–5

Feng X *et al* 2018 *Opt. Commun.* 410, 674–9

Fichou D and Morlock G E 2018 *Anal. Chem.* 90 6984–91

Giovannetti V, Guha S, Lloyd S, Maccone L, Shapiro J H and Yuen H P 2004 *Phys. Rev. Lett.* **92** 027902

Gondara L 2016 *Data Mining Workshops (ICDMW), 2016 IEEE 16th Int. Conf. on IEEE* pp 241–6

Grosshans F, Assche G V, Wenger J, Brouri R, Cerf N J and Grangier P 2003 *Nature* 421 238

Grosshans F and Grangier P 2002 *Phys. Rev. Lett.* 88 057902

Helstrom C W 1969 *J. Stat. Phys.* 1 231–52

Izumi S, Takeoka M, Fujiwara M, Pozza N D, Assalini A, Ema K and Sasaki M 2012 *Phys. Rev.* A 86 042328

Jang B, Chang J, Park A and Wu H 2018 *Radiat. Oncol.* 127 S1149

Kaushal H and Kaddoum G 2017 *IEEE Commun. Surv. Tutor.* 19 57–96

Ke X, Yang S and Wang J 2018 *2018 10th Int. Conf. on Advanced Infocomm Technology (ICAIT)* pp 61–5

Khan F N, Zhong K, Zhou X, Al-Arashi W H, Yu C, Lu C and Lau A P T 2017 *Opt. Express* 25 17767–76

Kulin M, Kazaz T, Moerman I and de Poorter E 2017 arXiv:(1712.03987)

Li Z, Meier M A, Hauksson E, Zhan Z and Andrews J 2018 *Geophys. Res. Lett.* **45** 4773–9

Lohani S and Glasser R T 2018 *Opt. Lett.* **43** 2611–14

Lohani S and Glasser R T 2019 *Frontiers in Optics 2019* p FTu5B–3

Lohani S, Knutson E M, O'Donnell M, Huver S D and Glasser R T 2018 *Appl. Opt.* **57** 4180–90

Loock P V 2011 *Laser & Photonics Reviews* **5** 167–200

Patnaik B and Sahu P 2013 *Int. Journal of Signal & Imaging Systems Engineering* **6** 3–8

Proakis J G and Salehi M 2001 *Digital Communications* (New York: McGraw-Hill) vol 4

Sanchez-Lengeling B and Aspuru-Guzik A 2018 *Science* **361** 360–5

Torlai G, Mazzola G, Carrasquilla J, Troyer M, Melko R and Carleo G 2018 *Nat. Phys.* **14** 447

Vincent P, Larochelle H, Bengio Y and Manzagol P A 2008 *Proc. of the 25th Int. Conf. on Machine Learning ACM* pp 1096–103

Wang D, Zhang M, Li J, Li Z, Li J, Song C and Chen X 2017 *Opt. Express* 25 17150–66

Wiseman H M and Milburn G J 2009 *Quantum Measurement and Control* (Cambridge: Cambridge University Press)

Yoshida M, Omiya T, Kasai K and Nakazawa M 2011 *Optical Fiber Communication Conf. 2011* p OTuN3

Zhang J, Chen W, Gao M, Ma Y, Zhao Y, Chen W and Shen G 2018 *Opt. Express* **26** 18684–98