MICAS 913: Deep Learning Theory

Simulation Project

Deep Learning of a Communication System

Mansoor Yousefi

Télécom Paris, Palaiseau, France

2022 – 2023

Deep learning of a nonlinear communication system

Linear channel

$$Y = X + N$$

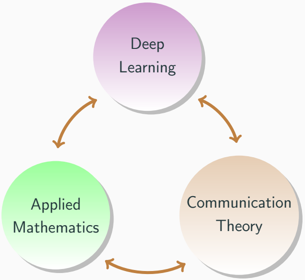
Easy?!

Nonlinear channel

$$Y = f(X, N)$$

Hard!

Inter-disciplinary project



Generative model

We first describe a **generative model**, to realize the data sets.

A generative model is a conditional probability distribution function (PDF)

$$p(\boldsymbol{y}|\boldsymbol{x}), \quad \boldsymbol{x}, \boldsymbol{y} \in \mathbb{C}^n.$$

We compute this function with a **multi-layer convolutional neural network (CNN)**.

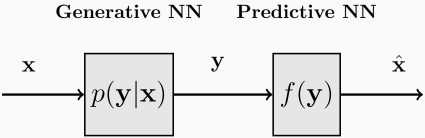
Predictive model

We will next design a **predictive model**, for inference on a $\boldsymbol{y} \sim p(\boldsymbol{y}|\boldsymbol{x})$.

This model could be a conditional PDF $q(\boldsymbol{x}|\boldsymbol{y})$ computed by a NN, followed by the maximum likelihood estimator

$$\begin{aligned}\hat{\boldsymbol{x}} &= \operatorname*{argmax}_{\boldsymbol{x}} q(\boldsymbol{x}|\boldsymbol{y}) \\ &\triangleq f(\boldsymbol{y}).\end{aligned}$$

Alternatively, the NN could compute the function $\hat{\boldsymbol{x}} = f(\boldsymbol{y})$.



The generative model has fixed weights and biases.

Given a training data set $\left\{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\right\}_{i=1}^{n}$, the predictive model is learned via ERM

$$f^* = \underset{f \in \mathcal{H}}{\operatorname{argmin}}\, \hat{R}(f)$$

where

$$\begin{aligned}
\hat{R}(f) &= \frac{1}{n} \sum_{i=1}^{n} \left\| \hat{\boldsymbol{x}}^{(i)} - \boldsymbol{x}^{(i)} \right\|_2^2 \\
&= \frac{1}{n} \sum_{i=1}^{n} \left\| f(\boldsymbol{y}^{(i)}) - \boldsymbol{x}^{(i)} \right\|_2^2
\end{aligned}$$

We consider a parameterized family

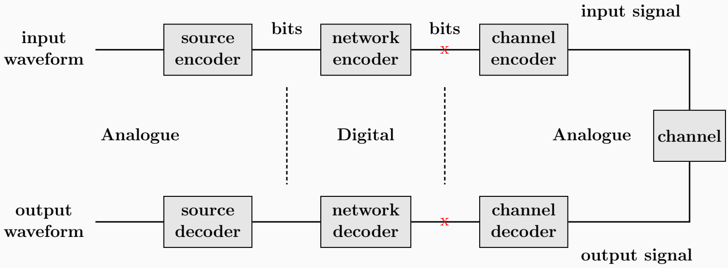
$$\mathcal{H} \triangleq \left\{ f_{\boldsymbol{w}}(\boldsymbol{y}) : \ \boldsymbol{w} \in \mathbb{R}^{M} \right\}$$

for a hyper-parameter $M \in \mathbb{N}$, implemented by NNs.

A point-to-point nonlinear communication system

- A simplified point-to-point nonlinear communication system
- Channel model
- Generative neural network

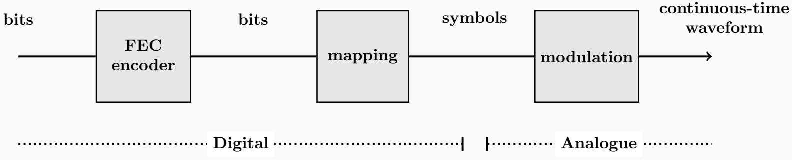
Block diagram of a communication system



Digital interfaces and layering: the *separation theorem* for point-to-point data communication implies that the source and channel can be designed separately.

We thus ignore the source and network encoder and decoder, and only learn the channel part. The other components can be learned separately.

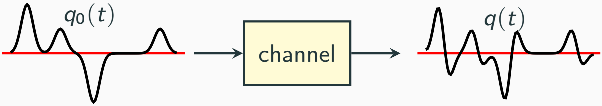
Channel encoder



- Forward error correction (FEC) encoder, to protect against channel noise
- Mapping: symbols to bits
- Modulation: symbols to signal

We ignore FEC as well (could be learned).

Channel model



- *A linear channel with additive noise*

$$Y(t) = h(t) * X(t) + N(t)$$

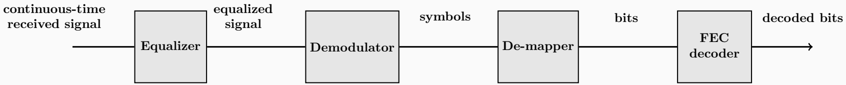
- *A complex channel described by a nonlinear PDE*. Here the signal evolves according to an evolution equation in 1+1 dimensions (time t, distance z)

$$\frac{\partial q}{\partial z} = K(q(t,z)) + n(t,z)$$

Examples: ($q_t := \partial_t q$)

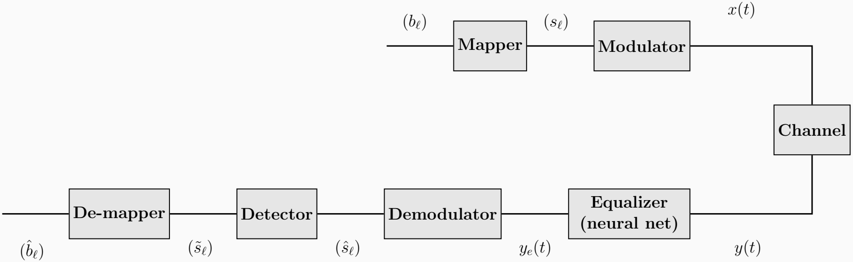
- $K(q) = |q|^2$ (memoryless nonlinearity)
- $K(q) = -j(q_{tt} + 2|q|^2 q)$ (nonlinearity with memory)

Channel decoder



- Equalizer: invert the deterministic effects in the channel
- Demodulator: signal to symbols mapping
- Demapper: bits to symbols
- FEC decoder: decoding for the FEC (neglected)

Eliminating blocks that we ignored, in this project we consider the following simplified block diagram.



In what follows, we describe each building block.

Linear Modulation

Let $\mathcal{H}$ be a separable *Hilbert space* of signals $x : \mathbb{R} \mapsto \mathbb{C}$ at the input of the channel with an inner product denoted by $\langle ., . \rangle$.

Let $\{\phi_\ell(t)\}_{\ell=-\infty}^{\infty}$ be an *orthonormal basis* for $\mathcal{H}$, *i.e.*, satisfying:

$$\langle \phi_i(t), \phi_j(t) \rangle = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$$

Linear Modulation

Linear modulation consists of expanding a signal $x(t) \in \mathcal{H}$ onto an orthonormal basis:

$$\boxed{x(t) = \sum_{\ell=-\infty}^{\infty} s_\ell \phi_\ell(t),}$$

where $\{s_\ell\}$ are symbols chosen from a constellation $\mathcal{C}$.

Example. Consider the space of T-periodic bounded signals, with the inner product

$$\langle f(t), g(t) \rangle = \frac{1}{T} \int_0^T f(t) g^*(t) \mathrm{d}t.$$

The *Fourier series* provides a choice of basis for $\mathcal{H}$:

$$\left\{ e^{jl\omega_0 t} \right\}_{l=-\infty}^{\infty}, \quad \omega_0 = \frac{2\pi}{T}.$$

$\square$

Example. Consider the space of finite-energy signals bandlimited to B Hz, denoted by $\mathcal{H} = L_B^2(\mathbb{R})$, with the inner product

$$\langle f(t), g(t) \rangle = \int_{-\infty}^{\infty} f(t) g^*(t) \mathrm{d}t.$$

The *Nyquist-Shannon sampling theorem* provides a choice of the basis:

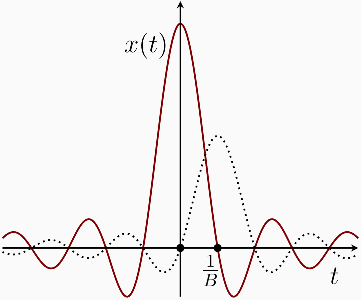
$$\left\{\sqrt{B}\,\mathsf{sinc}(Bt - \ell)\right\}_{\ell=-\infty}^{\infty}.$$

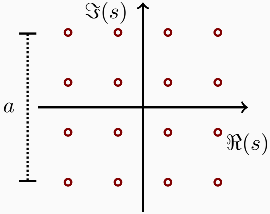
where $\mathsf{sinc}(x) = \sin(x)/(\pi x)$. A modification of this basis is the root raised cosines.

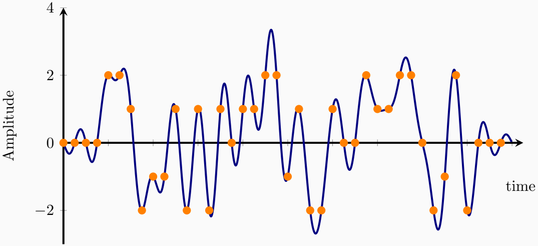
$\square$

In what follows, we consider $L_B^2(\mathbb{R})$.

An example of a linearly-modulated signal using a sinc pulse shape and a multi-ring constellation is shown here.







Considering the sinc basis, the transmitted signal is:

$$\boxed{x(t) = \sum_{\ell=-\infty}^{\infty} s_\ell \operatorname{sinc}(Bt - \ell)},$$

where s_ℓ are symbols drawn iid from $\mathcal{C}$.

It is shown in the project guide that the transmitter is subject to the average power constraint

$$\lim_{T\to\infty} \frac{1}{T} \mathsf{E}\left\{ \int_{-\frac{T}{2}}^{\frac{T}{2}} |x(\tau)|^2 \mathrm{d}\tau \right\} \leqslant \mathcal{P},$$

if and only if $\boxed{\mathsf{E}|s_k|^2 \leqslant \mathcal{P}}$. The pulse-shape is unit power (rather than unit energy).

Continuous-time channel model

Consider first a simple *additive white Gaussian noise (AWGN)* channel,

$$\boxed{y(t) = h(t) * x(t) + n(t)}\,.$$

Here, $x(t)$ is input signal, $y(t)$ is output signal, $h(t)$ is channel impulse response, $n(t)$ is white circular symmetric complex Gaussian noise, *i.e.*, satisfying

$$\mathsf{E}\Big\{n(t)n^*(t')\Big\} = \sigma_0^2\delta(t - t'),$$

where σ_0^2 is the noise power spectral density (PSD), and * denotes convolution. All signals are considered functions from $\mathbb{R}$ to $\mathbb{C}$.

The channel filter $h(t)$ introduces a distortion called **inter-symbol interference (ISI)**, that must be cancelled via **equalization**.

Note. Recall that the convolution of two functions $f(t)$ and $g(t)$ is defined as

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(t-\tau)g(\tau)\mathrm{d}\tau.$$

Further, the Fourier transform of $f \in L^1(\mathbb{R})$ is

$$\begin{aligned} \hat{f}(\omega) &= \int_{-\infty}^{\infty} f(t)e^{j\omega t}\mathrm{d}t, \\ &\triangleq \mathcal{F}(f)(\omega), \end{aligned}$$

where $\mathcal{F}$ denotes the Fourier transform operator.

Equalization

In the frequency domain, the AWGN channel is:

$$\hat{y}(\omega) = \hat{h}(\omega) \times \hat{x}(\omega) + \hat{n}(\omega),$$

where, *e.g.*, $\hat{x}(\omega) = \mathcal{F}(x)(\omega)$.

Equalization is performed in the frequency domain to invert the channel

$$\begin{aligned} \hat{y}_e(\omega) &\triangleq \hat{h}^{-1}(\omega)\hat{y}(\omega) \\ &= \hat{x}(\omega) + \hat{z}(\omega), \end{aligned} \tag{1}$$

where $\hat{z}(\omega) = \hat{h}^{-1}(\omega)\hat{n}(\omega)$, and we assumed $\hat{h}(\omega) \neq 0$, $\forall|\omega| \leqslant \pi B$.

The signal in time domain after equalization is

$$\begin{aligned} y_e(t) &= \mathcal{F}^{-1}(\hat{y}_e(\omega)) \\ &= x(t) + z(t). \end{aligned} \tag{2}$$

De-modulation

The received symbols at the output are obtained by **projection** onto the basis

$$\hat{s}_\ell = \frac{\langle y_e(t), \mathrm{sinc}(Bt-\ell)\rangle}{\langle \mathrm{sinc}(Bt-\ell), \mathrm{sinc}(Bt-\ell)\rangle} = B\int_{-\infty}^{\infty} y_e(t)\,\mathrm{sinc}(Bt-\ell)\mathrm{d}t. \tag{3}$$

The projection step is matched filtering.

Discrete-time model

Substituting (**??**) into (**??**), the continuous-time model (**??**) after equalization is discretized to the discrete-time model

$$\hat{s}_\ell = s_\ell + z_\ell, \quad \ell = 1, 2, \cdots .$$

It is shown in the project guide that, for the $\hat{h}$ that we consider in the project, $z_\ell \sim \ iid\ N(0, \sigma^2)$, where $\sigma^2 = \sigma_0^2 B$ is the noise variance.

- Continuous-time model: $x(t) \mapsto y(t)$
- Discrete-time model: $s_\ell \mapsto \hat{s}_\ell$

Detection

In general, $\hat{s}_\ell \notin \mathcal{C}$. Given $\hat{s}_\ell$, we shall map $\hat{s}_\ell$ to a symbol $\tilde{s}_\ell \in \mathcal{C}$.

The optimal receiver is the *maximum likelihood detector* applied to the conditional probability distribution $p(\hat{s}_\ell | s_\ell)$, *i.e.*,

$$\tilde{s}_\ell = \mathrm{argmax}_{s_\ell \in S} \;\; p(\hat{s}_\ell | s_\ell).$$

Since

$$p(\hat{s}_\ell | s_\ell) = \frac{1}{\pi\sigma^2} e^{-\frac{|\hat{s}_\ell - s_\ell|^2}{\sigma^2}}$$

maximizing $p(\hat{s}_\ell | s_\ell)$ amounts to minimizing the distance in the exponent. The maximum likelihood detector is simplified to the **minimum distance decoder**.

Given $\hat{s}_\ell$:

$$\tilde{s}_\ell = \mathrm{argmin}_{s_\ell \in \mathcal{C}} \ \left| s_\ell - \hat{s}_\ell \right|^2.$$

A more complex channel

The design of the transmitter TX and receiver RX discussed above relied on the **linearity property** of the channel. For cases where the model is known and is tractable, the use of deep learning is questionable. In fact, good receivers are known for the AWGN channel.

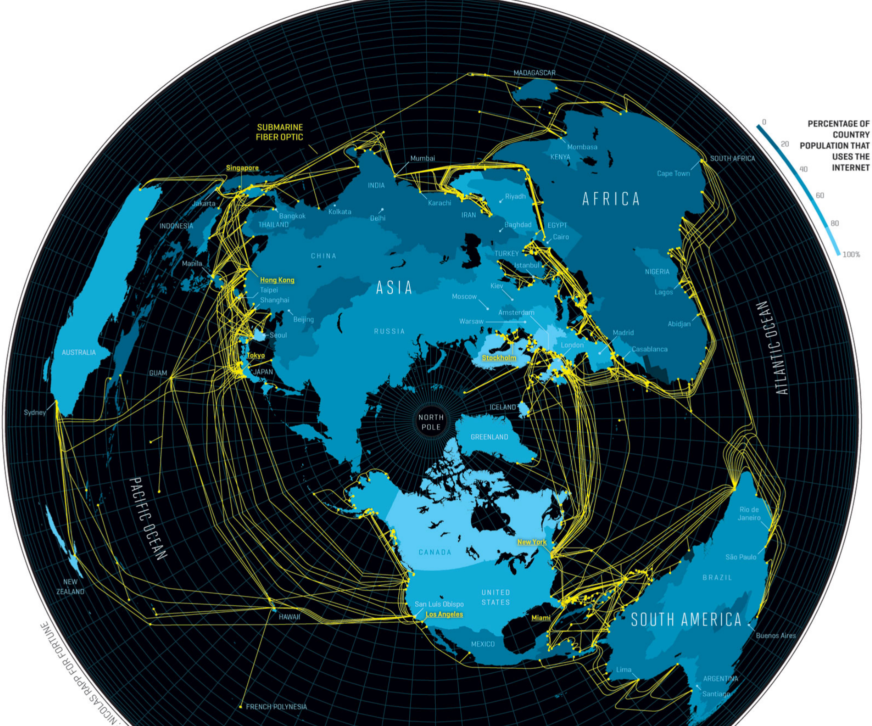
We thus consider a complex nonlinear channel where the optimal receiver is unknown. This channel is optical fiber, and takes the form:

$$\begin{aligned} Y &= f(X, N) \\ &\neq X + N. \end{aligned}$$

Project: End-to-end deep learning of the TX and RX for the nonlinear optical fiber using an input output data set.

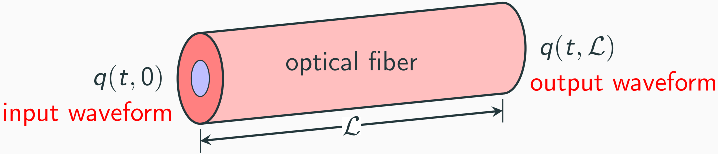
End-to-end indicates, from the input bits to the output bits; see the diagram.

Channel model



SUBMARINE FIBER OPTIC
PERCENTAGE OF COUNTRY POPULATION THAT USES THE INTERNET
0
20
40
60
80
100%
MADAGASCAR
Mombasa
KENYA
SOUTH AFRICA
Cape Town
AFRICA
Mumbai
Singapore
INDIA
Riyadh
Karachi
Jakarta
Bangkok
Kolkata
Delhi
IRAN
INDONESIA
THAILAND
Baghdad
EGYPT
Cairo
CHINA
TURKEY
Istanbul
NIGERIA
Manila
Hong Kong
Taipei
Shanghai
ASIA
Kiev
Moscow
Lagos
Beijing
Amsterdam
Warsaw
Abidjan
Seoul
RUSSIA
Madrid
London
Casablanca
AUSTRALIA
Tokyo
Stockholm
JAPAN
GUAM
ATLANTIC OCEAN
NORTH POLE
ICELAND
Sydney
GREENLAND
PACIFIC OCEAN
Rio de Janeiro
New York
CANADA
São Paulo
NEW ZEALAND
BRAZIL
UNITED STATES
San Luis Obispo
Los Angeles
Miami
SOUTH AMERICA
HAWAII
Buenos Aires
MEXICO
Lima
ARGENTINA
Santiago
FRENCH POLYNESIA
NICOLAS RAPP FOR FORTUNE

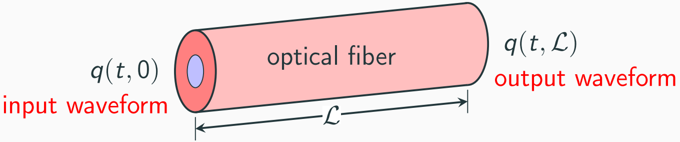
Optical Fiber



Desirable properties of **optical fiber**:

- Low loss $\sim$ 0.2 dB/km
- Large bandwidth $\sim$10's THz
- Immune to external interference
- Cost-effective

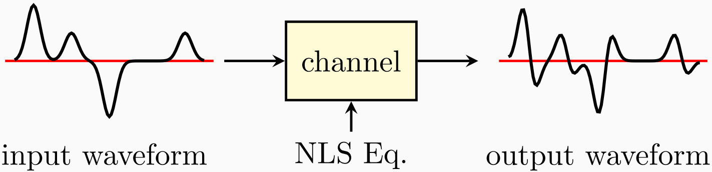
Stochastic Nonlinear Schrödinger Equation



Pulse propagation in optical fiber is modeled by the *stochastic nonlinear Schrödinger (NLS) equation*:

$$\frac{\partial q(t,z)}{\partial z} = \underbrace{-\frac{j\beta_2}{2}\frac{\partial^2 q(t,z)}{\partial t^2}}_{\textbf{dispersion}} + \underbrace{j\gamma|q(t,z)|^2 q(t,z)}_{\textbf{nonlinearity}} + \underbrace{n(t,z)}_{\textbf{noise}}$$

- *z is distance* along the fiber and *t is time*
- $q(t,z)$ is the complex envelope of the propagating signal
- β_2 is the second-order *chromatic dispersion coefficient*
- γ is the *nonlinearity parameter*
- $n(t,z)$ is bandlimited *white circular symmetric Gaussian noise*



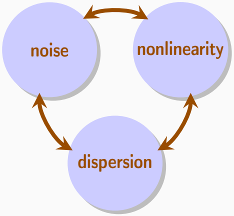
The input output of the channel are respectively

$$x(t) = q(t, 0), \quad y(t) = q(t, \mathcal{L}),$$

where $\mathcal{L}$ is fiber length.

The model may be generalized to include additional effects, such as residual loss, higher-order dispersion, multiple input multiple output transmission, etc.

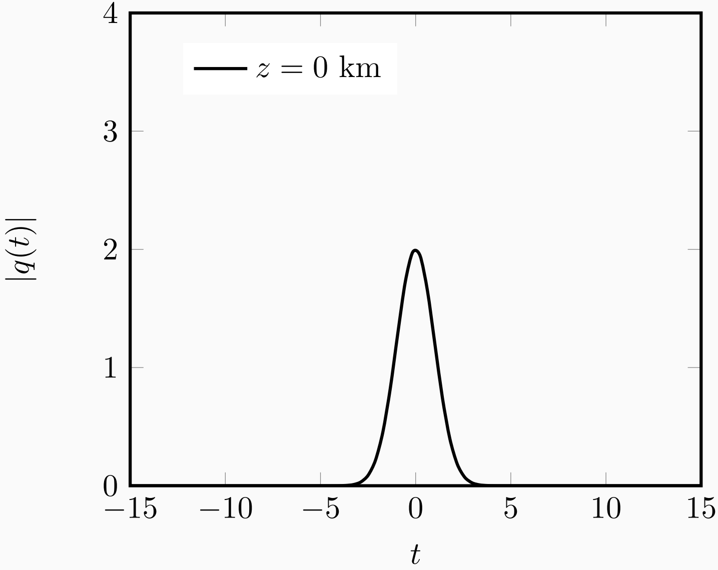
Physical Effects in the Channel



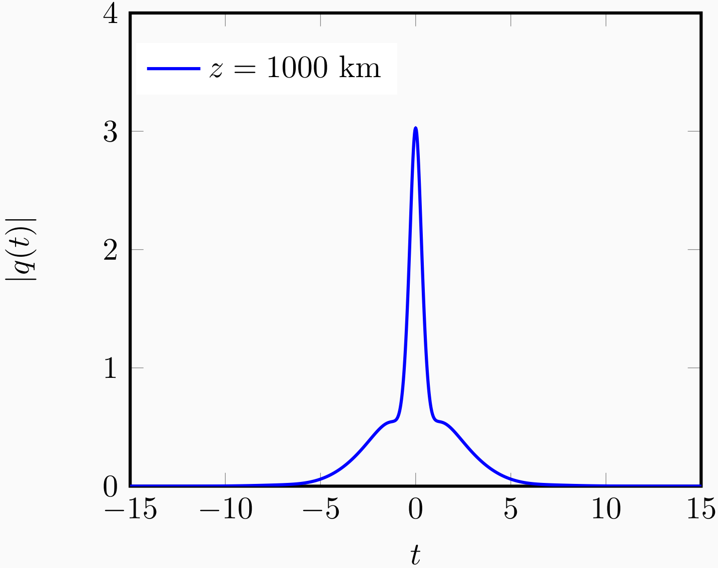
The model is governed by a *nonlinear dispersive partial differential equation (PDE)*, modeling the physical effects of **dispersion**, **nonlinearity** and **noise**.

This PDE cannot be solved analytically.

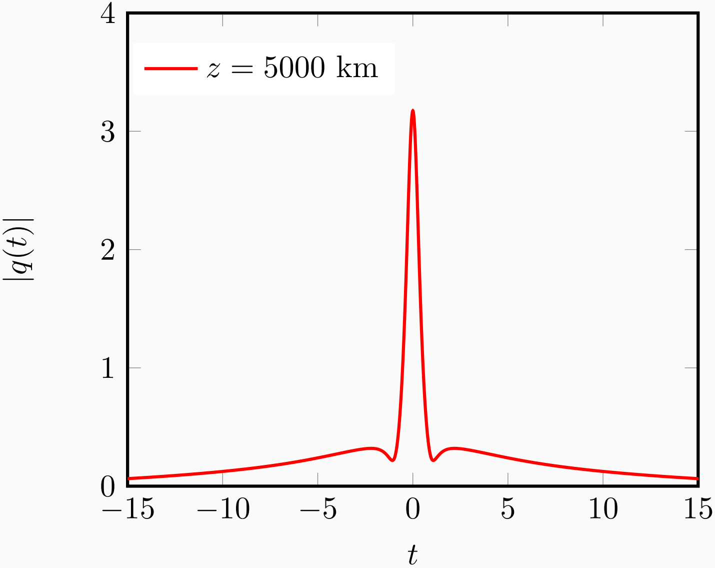
Example of Signal Propagation – a Gaussian input



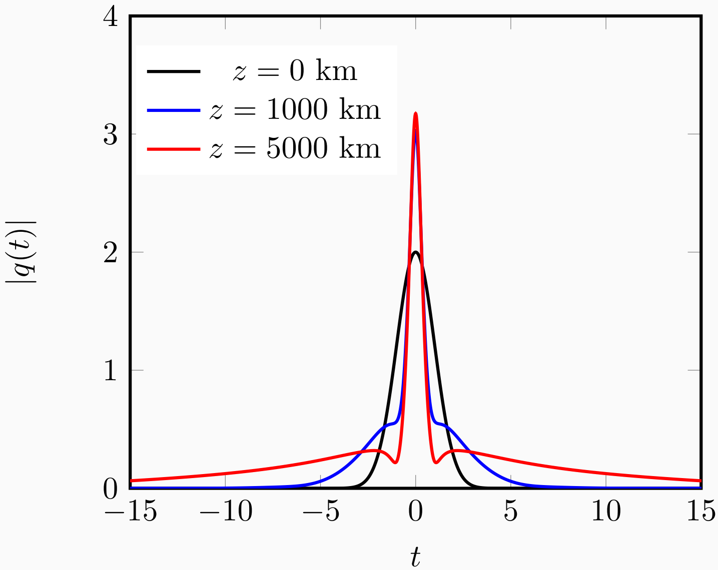
Example of Signal Propagation – a Gaussian input



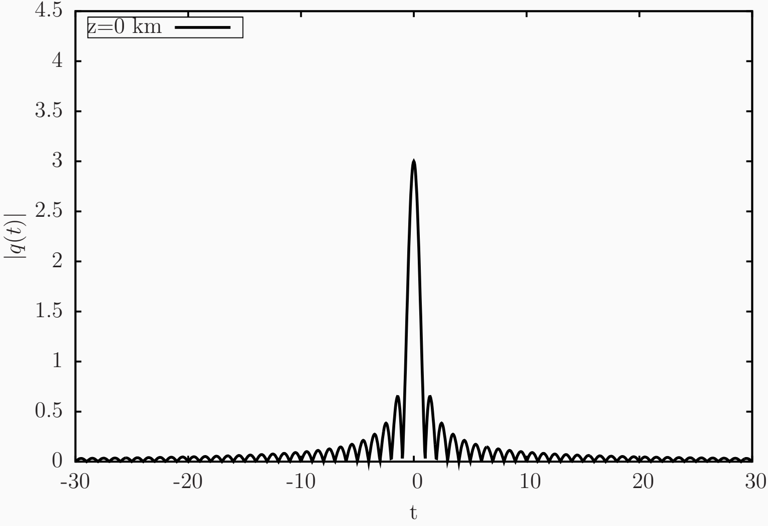
Example of Signal Propagation – a Gaussian input



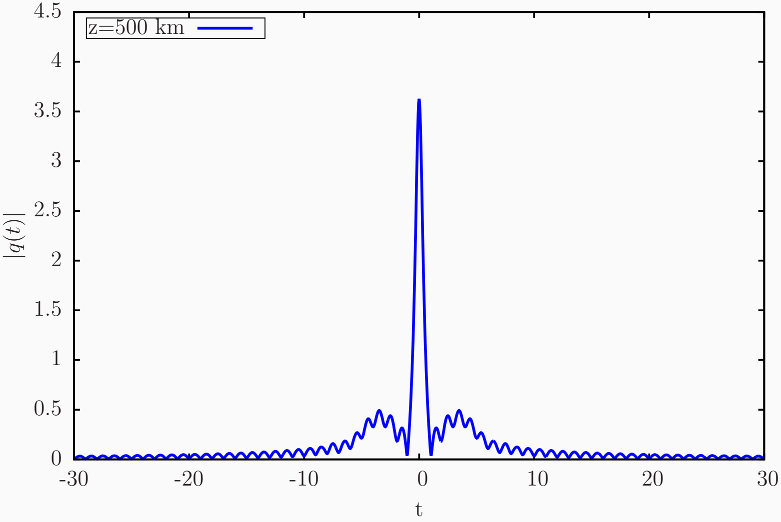
Example of Signal Propagation – a Gaussian input



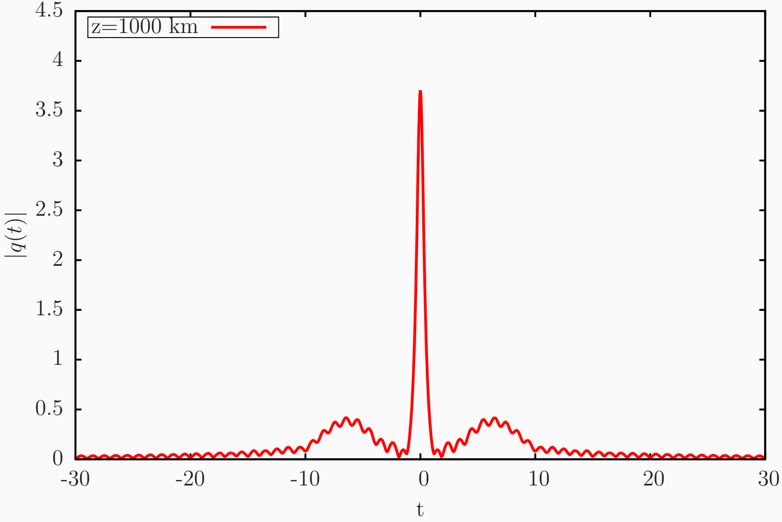
Example of Signal Propagation – a sinc input



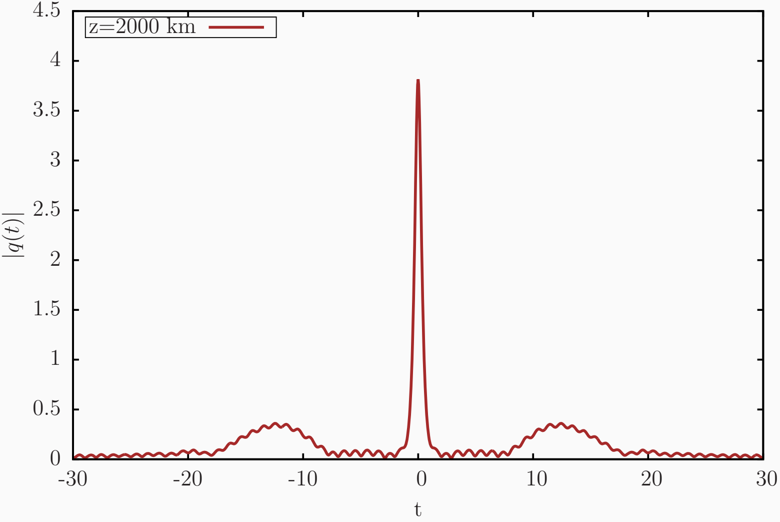
Example of Signal Propagation – a sinc input



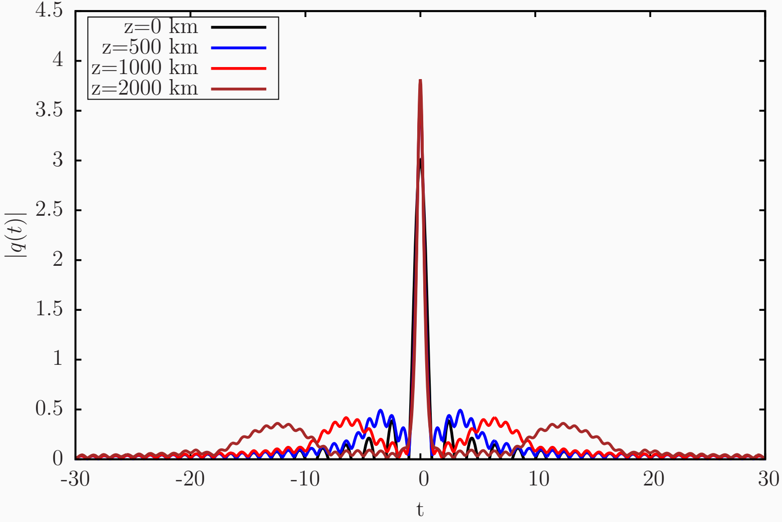
Example of Signal Propagation – a sinc input



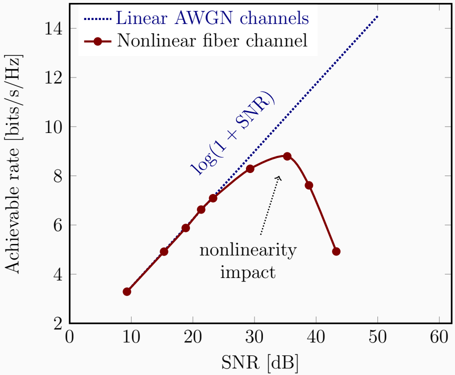
Example of Signal Propagation – a sinc input



Example of Signal Propagation – a sinc input



Achievable Information Rates (AIRs)



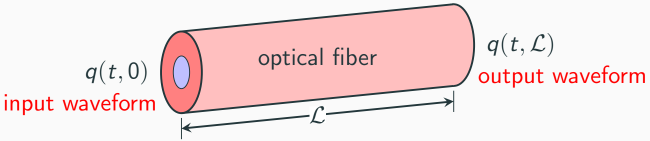
Fiber nonlinearity limits the AIRs of the current receivers

Can neural nets learn a better (or even optimal) receiver, and improve the AIRs?

We now describe the **generative neural network**. This model has an architecture that is based on the *split-step Fourier method (SSFM)*, a method for solving nonlinear dispersive PDEs.

We begin by discretizing the nonlinear PDE.

Generative neural network



Consider a general model:

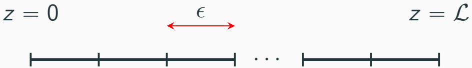
$$\frac{\partial q(t,z)}{\partial z} = \underbrace{L_L(q)}_{\textbf{linear}} + \underbrace{L_N(q)}_{\textbf{nonlinear}} + \underbrace{n(t,z)}_{\textbf{noise}}$$

where L_L and L_N are linear and nonlinear operators.

Example. In the stochastic nonlinear Schrödinger equation:

$$L_L(q) = -\frac{j\beta_2}{2}\frac{\partial^2 q}{\partial t^2},$$
$$L_N(q) = j\gamma|q|^2 q.$$

Split-Step Fourier Method



Discretize the distance into a large number $M \to \infty$ of segments of length $\epsilon = \mathcal{L}/M$.

Discretize the time interval $[-T/2, T/2]$ into a large number $n \to \infty$ of intervals of step size $\mu = T/n$.

Break down the PDE into 3 parts. Perform successive linear, nonlinear and noise transformations in distance, in each spatial segment:

- **Linear step**:

$$\partial_z q = L_L(q) \quad \Rightarrow \quad \boldsymbol{V} = W\boldsymbol{X}$$

where $\boldsymbol{X} \in \mathbb{C}^n$, $\boldsymbol{V} \in \mathbb{C}^n$ and $W \in \mathbb{C}^{n\times n}$ is a weight matrix.

- **Nonlinear step**:

$$\partial_z q = L_N(q) \quad \Rightarrow \quad \boldsymbol{U} = \sigma(\boldsymbol{V})$$

where $\boldsymbol{U} \in \mathbb{C}^n$, and $\sigma(x) : \mathbb{C} \mapsto \mathbb{C}$ is the activation function acting per-component.

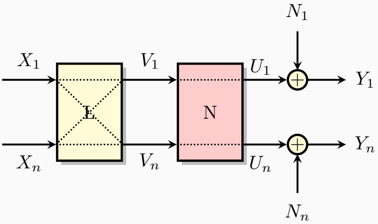
- **Noise addition:**

$$\partial_z q = n(t, z) \quad \Rightarrow \quad \boldsymbol{Y} = \boldsymbol{U} + \boldsymbol{N},$$

where $\boldsymbol{N} \sim N(0, \sigma^2 I_n)$.

Yellow and blue colors indicate the continuous- and discrete-time models, respectively.

One layer of the neural network



The diagram depicts one layer of the neural network, consisting of a linear, a nonlinear and a noise transformation. The nnet will consist of a cascade of at least 100 fully-connected layers!

Note. In the NN used in the inference mode, one performs successive linear and nonlinear transformations. There is no noise step. In the generative NN, there exists a noise step. The data set is obtained from a conditional PDF $p(\boldsymbol{Y}|\boldsymbol{X})$, not a deterministic function. $\square$

Linear Step

$$\frac{\partial q(t,z)}{\partial z} = -\underbrace{\frac{j\beta_2}{2}\frac{\partial^2 q(t,z)}{\partial t^2}}_{\textbf{dispersion}}$$

Define the Fourier transform with respect to t with convention:

$$\hat{q}(\omega, z) = \mathcal{F}(q)(\omega) = \int_{-\infty}^{\infty} q(t,z)e^{-j\omega t}\mathrm{d}t.$$

Taking Fourier transform of both sides:

$$\frac{\partial \hat{q}(\omega, z)}{\partial z} = \frac{j\omega^2\beta_2}{2}\hat{q}(\omega, z) \quad \Rightarrow \begin{cases} \hat{q}(\omega, z) = e^{j\frac{\beta_2 z}{2}\omega^2}\hat{q}(\omega, 0) & (*) \\ q(t,z) = \frac{1}{\sqrt{j2\pi\beta_2 z}}e^{j\frac{1}{2\beta_2 z}t^2} * q(t,0) \end{cases}$$

where we used $e^{-\frac{t^2}{2\lambda}} \leftrightarrow \sqrt{2\pi\lambda}e^{-\frac{\lambda\omega^2}{2}}$.

In the frequency domain, dispersion is a simple phase change (an all-pass filter).

In the time domain, dispersion is a convolution, giving rise to memory, pulse broadening and ISI

Let the input output of the linear step in one layer be $\boldsymbol{X} \in \mathbb{C}^n$ and $\boldsymbol{V} \in \mathbb{C}^n$, respectively. Discretizing (*),

$$\boldsymbol{V} = W\boldsymbol{X}$$

where $W = D^H \Gamma D$, where D is the n-point discrete Fourier transform (DFT) matrix at frequencies $(\omega_i)_{i=1}^n$. Further, $\Gamma = \mathrm{diag}(\hat{\boldsymbol{h}})$ where $\hat{\boldsymbol{h}} = [e^{j\frac{\beta_2 \epsilon}{2}\omega_1^2}, \cdots, e^{j\frac{\beta_2 \epsilon}{2}\omega_n^2}]$. Let

$$\boldsymbol{h} = \mathrm{IDFT}(\hat{\boldsymbol{h}}).$$

Linear step is convolution of the input with the dispersion filter $\boldsymbol{h}$. The weight matrix is W, which is Toeplitz with the first row $\boldsymbol{h}$.

Note. In generative model, the filter $\boldsymbol{h}$ is fixed, and not learned. In inference NN, $\boldsymbol{h}$ must be learned. □

Nonlinear Step

$$\frac{\partial q(t,z)}{\partial z} = \underbrace{j\gamma |q(t,z)|^2 q(t,z)}_{\textbf{nonlinearity}}$$

Set $q(t,z) = r(t,z)e^{j\phi(t,z)}$. We have

$$(\dot{r} + jr\dot{\phi})e^{j\phi(t,z)} = j\gamma r^3 e^{j\phi(t,z)}$$

where dot denotes z derivative. Then:

$$\dot{r} + jr\dot{\phi} = j\gamma r^3 \qquad \Rightarrow \qquad \begin{cases} \dot{r} = 0 & \Rightarrow \quad r(t,z) = r(t,0) \\ \dot{\phi} = \gamma r^2 & \Rightarrow \quad \phi(t,z) = \phi(t,0) + \gamma z r^2(t,0) \end{cases}$$

Thus:

$$q(t,z) = r(t,z)e^{j\phi(t,z)} = r(t,0)e^{j\phi(t,0) + j\gamma z r^2(t,0)} = q(t,0)e^{j\gamma |q(t,0)|^2 z}$$

$$
\begin{cases}
q(t,z) = q(t,0)e^{j\gamma|q(t,0)|^2 z}, & (**) \\
\hat{q}(\omega,z) = \hat{q}(\omega,0) * \mathcal{F}\left(e^{j\gamma|q(t,0)|^2 z}\right)(\omega)
\end{cases}
$$

In the time domain, nonlinearity is a simple signal-dependent phase shift

In the frequency domain, nonlinearity is a convolution, giving rise to spectral broadening and interference

Note that $(**)$ is memoryless. Let the input output of the nonlinear step in one layer be $\boldsymbol{V} \in \mathbb{C}^n$ and $\boldsymbol{U} \in \mathbb{C}^n$, respectively.

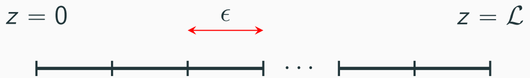
Discretizing (**)

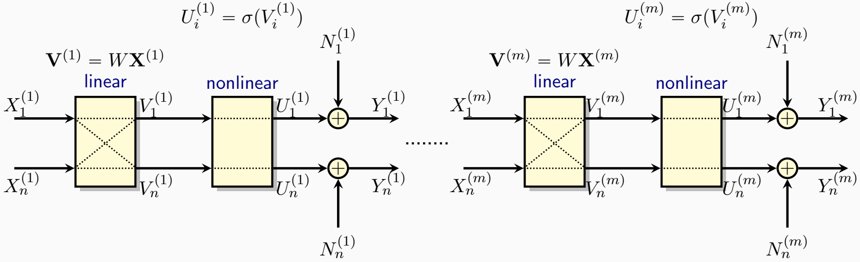
$$U_i = \sigma(V_i),$$

where $\sigma(x)$ is the **activation function**

$$\boxed{\sigma(x) = xe^{j\gamma\epsilon|x|^2}}.$$

Generative Neural Network





- **Linear transformation**:

$$\boxed{\boldsymbol{V}^{(\ell)} = \mathrm{W}\boldsymbol{X}^{(\ell)}}$$

where W is the weight matrix.

- **Nonlinear transformation**:

$$\boxed{U_i^{(\ell)} = \sigma(V_i^{(\ell)})},$$

where $\sigma(x) = xe^{j\epsilon\gamma|x|^2}$ is the activation function.

- **Noise addition:**

$$\boxed{\boldsymbol{Y}^{(\ell)} = \boldsymbol{U}^{(\ell)} + \boldsymbol{N}^{(\ell)}}, \quad \boldsymbol{N}^{(\ell)} \sim iid\ N(0, \sigma^2 I_n)$$

The iteration for the vector propagating in the NN is

$$\boldsymbol{Y}^{(\ell+1)} = \sigma\big(\mathsf{W}\boldsymbol{Y}^{(\ell)}\big) + \boldsymbol{N}^{(\ell)}, \quad \ell = 0, 1, \cdots, m-1,$$

initialized with the input vector $\boldsymbol{Y}^{(0)} = \boldsymbol{X}$.

References

Digital communication

[1] A. Lapidoth, A foundation in digital communication, Cambridge University Press, 2017.

[2] R. G. Gallager, Principles of Digital Communication, Cambridge University Press, 2008.

[3] D. Forney, 6.451 Principles of Digital Communication II, MIT OpenCourseWare, Spring 2005.

[4] D. Forney and G. Ungerboeck, "*Modulation and coding for linear Gaussian channels*," IEEE Trans. Info. Theory, vol. 44, no. 6, pp. 2384–2415, 1998.

Deep learning, experimental approach

[5] I. Goodfellow and Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.

Split-step Fourier method, information theory of fiber

[6] G. Kramer, M. I. Yousefi, and F. Kschischang, "*Upper bound on the capacity of a cascade of nonlinear and noisy channels*," in IEEE Info. Theory Workshop, Jerusalem, Israel, Apr. 2015.

[7] R. J. Essiambre, G. Kramer, P. J. Winzer, G. J. Foschini, and B. Goebel, "*Capacity limits of optical fiber networks*," IEEE J. Lightw. Technol., vol. 28, no. 4, pp. 662–701, Feb. 2010.

Deep learning of the fiber-optic channel

[8] C. Häger and H. D. Pfister, "*Deep learning of the nonlinear Schrödinger equation in fiber-optic communications*," ISIT 2018.

[9] R. M. Bütler, C. Häger, H. D. Pfister, G. Liga and A. Alvarado, "*Model-based machine learning for joint digital backpropagation and PMD compensation*," J. of Light. Technol., vol. 39, no. 4, pp. 949–959, 2020.

Reference 1, 5, 8, 9 are most directly applicable.

Temporary page!

LaTeX was unable to guess the total number of pages correctly. A
there was some unprocessed data that should have been added t
the final page this extra page has been added to receive it.

If you rerun the document (without altering it) this surplus page
will go away, because LaTeX now knows how many pages to expe
for this document.