Proceedings of Machine Learning Research vol 107:74–106, 2020

SchrödingeRNN: Generative Modeling of Raw Audio as a Continuously Observed Quantum State

Beñat Mencia Uranga BEINAT.MENCIA@GMAIL.COM
and **Austen Lamacraft** AL200@CAM.AC.UK
TCM Group, Cavendish Laboratory, University of Cambridge, J. J. Thomson Ave., Cambridge CB3 0HE, UK

Abstract

We introduce SchrödingeRNN, a quantum-inspired generative model for raw audio. Audio data is wave-like and is sampled from a continuous signal. Although generative modeling of raw audio has made great strides lately, relational inductive biases relevant to these two characteristics are mostly absent from models explored to date. Quantum Mechanics is a natural source of probabilistic models of wave behavior. Our model takes the form of a stochastic Schrödinger equation describing the continuous time measurement of a quantum system, and is equivalent to the *continuous Matrix Product State* (cMPS) representation of wavefunctions in one dimensional many-body systems. This constitutes a deep autoregressive architecture in which the system's state is a latent representation of the past observations. We test our model on synthetic data sets of stationary and non-stationary signals. This is the first time cMPS are used in machine learning.

Keywords: Machine Learning, Generative Models, Quantum Physics, Matrix Product States.

1. Introduction

Audio generation appears in different machine learning tasks such as music synthesis or *text-to-speech*, where the input is text and the output is speech audio. One of the reasons why it is challenging is that the dimensionality of the raw audio signal is usually a lot larger than that of the effective semantic-level signal. In speech synthesis for instance, one is typically interested in generating utterances corresponding to full sentences. At a minimum quality sampling rate of 16kHz, an average of 6,000 samples per word are generated Mehri et al. (2016).

Both music and speech are complex and highly structured. In audio signal form, different features have different timescales, ranging from milliseconds to minutes in the case of music. Because the correlations span different orders of magnitude, modeling the temporal correlations of the signals is challenging Dieleman et al. (2018).

Traditionally, the high dimensionality of the raw audio modelling problem has been dealt with by compressing the audio waveforms into spectral or higher level features, and then defining generative models on these features. Examples in music generation are symbolic representations such as scores and MIDI sequences. The compression causes many of the subtleties that are crucial for the quality of sound to vanish. A way around these limitations is to model sound in the raw audio domain instead. While the digital form of audio is also lossy, the relevant information for the quality of musicality is retained.

There has been recent work on raw audio modelling using autoregressive models: AMAE Dieleman et al. (2018), WaveNet van den Oord et al. (2016), VRNN Chung et al. (2015), WaveRNN Kalchbrenner et al. (2018) and SampleRNN Mehri et al. (2016). The first is a convolutional neural network with dilated convolutions, the rest are recurrent neural networks. Beyond autoregressive models, there is WaveGlow Prenger et al. (2018) where a flow model is used and WaveGAN Donahue et al. (2018) using generative adversarial networks.

2. Quantum-inspired machine learning

A natural connection between quantum mechanics and machine learning is that probability distributions appear in both disciplines. Quantum-inspired machine learning is the use of quantum wave functions and quantum processes to model probability distributions and generative processes. In each case, one needs to choose the wave function and the physical process that is suitable for the problem at hand.

In raw audio modeling, the data is wave-like and quantum mechanics is a natural source of probabilistic models of wave behaviour. Hence, quantum-inspired models might benefit from the *inductive bias* induced by these two characteristics: the wave-like and probabilistic nature. Furthermore, within the range of problems that exist in machine learning, one-dimensional machine learning is specially appealing for quantum many-body physicists. This is because in physics, the most powerful numerical and analytical tools have been developed to study one-dimensional systems. Therefore, there is the potential to use them to solve machine learning tasks. In this work, we use *continuous matrix product states* (cMPS) (see Appendix A), a numerical tool used in many-body quantum physics to deal with Hilbert spaces of many-body systems, to handle the high dimensionality of the audio data. There have been several works where *matrix product states* and more general tensor networks have been used for machine learning Glasser et al. (2019); Cheng et al. (2019); Stokes and Terilla (2019); Li and Zhang (2018); Han et al. (2017); Efthymiou et al. (2019); Liu et al. (2019); Evenbly (2019); Guo et al. (2018); Glasser et al. (2018); Stoudenmire (2018); Novikov et al. (2016); Bradley et al. (2019); Stoudenmire and Schwab (2016). RNNs have previously been used to learn the Schrödinger equation Wang et al. (2019). As far as we are aware, our work is the first where cMPS are used.

3. A quantum-inspired model for raw audio

In a typical raw audio dataset, each data point is a vector with several thousands of real valued elements, e.g. in NSynth dataset Engel et al. (2017) each note amounts to 64,000 samples. Hence, data lives in a very high dimensional space, which makes it unaffordable to explore brute-force: we are faced with *the curse of dimensionality*.

This is reminiscent of a problem that arises in many-body quantum optimization problems. When trying to find the variational ground state of a many-body quantum system, one has an exponentially large Hilbert space to explore. Matrix product states (MPS) serve as a tool to overcome the curse of dimensionality in this context. As explained in Orus (2014), it gives a way to parameterize the relevant corner of the Hilbert space efficiently.

The fact that MPS has proven to be a successful tool to overcome the curse of dimensionality in physics suggests that it might be useful in machine learning as well. In this work, we want to explore the utility of MPS to model raw audio. On the other hand, MPS is not suitable for modeling continuous data (like raw audio), because it describes lattices of discrete degrees of freedom like spins. As explained in Appendix A, there exists a generalization of MPS to systems with continuous degrees of freedom: *continuous matrix product states* (cMPS).

We will be thinking of the audio waveforms as the outcome of a sequential measurement of a continuous observable throughout the evolution of a quantum system.

3.1. The SchrödingeRNN model

As explained in Appendix B, our model generative process consists on the continuous measurement of the homodyne current I_t (see Appendix B.1), on the output of an open quantum system described by a cMPS. As a refinement of the cMPS model, we include two extra variables: A and σ. The model involves the signal I_t together with a latent Hilbert space consisting of vectors $|\psi\rangle \in \mathbb{C}^D$. The signal follows the stochastic process

$$I_{t+dt} = A\langle R_t + R_t^\dagger\rangle_t + z, \quad \text{where} \quad z \sim N(0, 1/dt). \tag{1}$$

The parameter A is a real learning variable, $R_t = e^{iHt} R e^{-iHt}$ (H is real and diagonal), $R \in \mathbb{C}^{D\times D}$ is a matrix acting on the latent space and as before the angular brackets $\langle\cdot\rangle_t$ denote the quantum mechanical expectation over an (unnormalized) state $|\tilde{\psi}_t\rangle$

$$\langle\cdot\rangle_t = \frac{\langle\tilde{\psi}_t|\cdot|\tilde{\psi}_t\rangle}{\langle\tilde{\psi}_t|\tilde{\psi}_t\rangle}. \tag{2}$$

The state $|\tilde{\psi}\rangle$ evolves according to

$$|\tilde{\psi}_{t+dt}\rangle = \left[\mathbb{1} - \frac{\sigma^2}{2} R_t^\dagger R_t dt + R_t I_{t+dt} dt\right] |\tilde{\psi}_t\rangle\,, \tag{3}$$

$$|\psi_{t+dt}\rangle = |\tilde{\psi}_{t+dt}\rangle \,/ \sqrt{\langle\tilde{\psi}_{t+dt}|\tilde{\psi}_{t+dt}\rangle}. \tag{4}$$

The purpose of introducing the training variable A is to learn the amplitude of the signal. The amplitude is set by $A\langle R_t + R_t^\dagger\rangle_t$ in Eq. (1). This is done to learn R independently of the amplitude of the signals in the dataset. This way, the training of R is geared solely towards optimizing the time evolution of $|\tilde{\psi}_t\rangle$ in Eq. (3). The hyperparameter dt sets the strength of the term $R_t^\dagger R_t$ compared to $R_t I_{t+dt}$ (one can see this by absorbing $\sqrt{dt}$ into R). In cases where we are interested in fixing dt to be the real time discretization of the data (see Sec. 6.5), σ is the hyperparameter in charge of this. The initial state $|\psi_0\rangle$ is learned.

The conditional joint probability density for a sequence of measurements $\{I_t\}$ is

$$p(I_T, ..., I_1|H, R, A, |\psi_0\rangle) = \prod_{k=0}^{T-1} p(I_{k+1}|I_k, ..., I_1; H, R, A, |\psi_0\rangle), \text{ where}$$

$$p(I_{k+1}|I_k, ..., I_1; H, R, A, |\psi_0\rangle) = \sqrt{\frac{dt}{2\pi}} \exp\left[-\frac{dt}{2}\left(I_{k+1} - A\langle R_k + R_k^\dagger\rangle_k\right)^2\right]. \tag{5}$$

This constitutes an *Autoregressive Recurrent Neural Network* where the hidden state is the quantum wavefunction $|\tilde{\psi}\rangle$ and the non-linear update equation is Eq. (4). Aside from a few last details that will be explained in the coming sections, this probability distribution defines our *quantum-inspired model*. A detailed derivation of the model can be found in Appendix B.

4. Advantages of the cMPS approach

As explained in the introduction, the traditional way of dealing with the high dimensionality of raw audio data has been to compress the audio waveforms into spectral or higher level features, and

then defining generative models on these features. As a consequence, many of the nuances that are essential for the quality of the sound vanish.

The alternative is to model raw audio waveforms directly. This approach consists in modeling the joint probability distribution of the dataset (the waveforms). There has been recent work using deep neural networks: Dieleman et al. (2018); van den Oord et al. (2016); Chung et al. (2015); Kalchbrenner et al. (2018); Mehri et al. (2016); Prenger et al. (2018); Donahue et al. (2018). Audio data is wave-like and is sampled from a continuous signal. Although generative modeling of raw audio has made great strides lately, relational inductive biases relevant to these two characteristics are mostly absent from models explored to date. The cMPS model introduces these inductive biases.

5. Data as homodyne current

We have seen that the sequential measurement of the homodyne current on the output of an open quantum system described by a cMPS gives rise to the autoregressive probability distribution shown in Eq. (5). We now want to use this probability distribution to model raw audio data x_t. One obvious thing to do is to consider the raw audio to be the homodyne current, i.e. $I_t \equiv x_t$, in which case the generative model is defined as

$$
\begin{aligned}
p(x_T, ..., x_1 | H, R, A, |\psi_0\rangle) &= \prod_{k=0}^{T-1} p(x_{k+1} | x_k, ..., x_1; H, R, A, |\psi_0\rangle), \\
p(x_{k+1} | x_k, ..., x_1; H, R, A, |\psi_0\rangle) &= \sqrt{\frac{dt}{2\pi}} \exp\left[-\frac{dt}{2} \left(x_{k+1} - A\langle R_k + R_k^\dagger \rangle_k \right)^2 \right].
\end{aligned} \tag{6}
$$

Note that in the limit $dt \to 0$, the variance $1/dt$ diverges and the samples of this probability distribution become pure noise. Hence, this model does not have a continuous limit in the sense that as the time discretization becomes dense, the signal does not become smoother but more discontinuous.

If our training strategy is *maximum log likelihood*, the loss function of a single data point is

$$
-\log p(x_T, ..., x_1 | H, R, A, |\psi_0\rangle) = -\sum_{k=0}^{T-1} \log p(x_{k+1} | x_k, ..., x_1; H, R, A, |\psi_0\rangle). \tag{7}
$$

Since dt is a hyperparameter (i.e., we do not learn it), we define the loss function as

$$
\text{loss}(H, R, A, |\psi_0\rangle) = \sum_{k=0}^{T-1} \left(x_{k+1} - A\langle R_k + R_k^\dagger \rangle_k \right)^2. \tag{8}
$$

At sampling time, the variance of the Gaussian in Eq. (6) is tuned by introducing a temperature parameter T (explained in Sec. 6.6) to optimize the quality of the samples. Therefore, dt does not influence the variance at generation time.

6. Time derivative of data as homodyne current: a stochastic differential equation perspective

Let us consider Eq. (1). Note that $z \sim N(0, q^2)$ is equivalent to $qz \sim N(0, 1)$ and therefore multiplying both sides by dt

$$I_{t+dt}dt = A\langle R_t + R_t^\dagger\rangle_t dt + d\beta_t. \tag{9}$$

The process β_t is Brownian motion and its independent increments have variance dt. In the limit $dt \to 0$, this equation is reminiscent of a *stochastic differential equation*

$$dI_t = \mathrm{f}(I_t, t)dt + d\beta_t. \tag{10}$$

On the other hand, the left hand side of Eq. (9) contains the value of the stochastic process I_t whereas the left hand side of Eq. (10) contains the differential of the process $dI_t \equiv I_{t+dt} - I_t$. In order to rephrase our model in the language of stochastic differential equations, an option is to define the time derivative of the raw audio data to be the outcome of the homodyne current measurement, i.e. $I_t \equiv dx_t/dt$, instead of $I_t \equiv x_t$.

As explained in Chen et al. (2018), there are several advantages of having a continuous formulation of the model, even though one always needs to discretize to perform numerical calculations. One of the main advantages is that one can use the machinery developed to numerically integrate stochastic differential equations.

Even though it is appealing to rephrase the model in terms of SDEs, later we will see that this is not always a good option, since for certain datasets, the time derivative dx_t/dt of the signals is more spiky and discontinuous than the signal x_t, which is problematic for training our model. In the remainder of the paper, we will use both approaches. Depending on the choice, the notation will be

- If $I_t \equiv x_t,$ then $x_{t+dt} = A\langle R_t + R_t^\dagger\rangle_t + z,$ where $z \sim N(0, 1/dt)$.
- If $I_t \equiv dx_t/dt,$ then $dx_t = A\langle R_t + R_t^\dagger\rangle_t dt + d\beta_t$.

Throughout the paper, unless we consider it helpful, we do not specify the units of different quantities.

6.1. The model from an SDE perspective

In the continuous formulation of the model, the signal follows the stochastic process (Itô process)

$$dx_t = A\langle R_t + R_t^\dagger\rangle_t dt + d\beta_t. \tag{11}$$

Here β_t is Brownian motion with diffusion constant q, which means that the independent increments $\Delta\beta \equiv \beta_{k+1} - \beta_k$ are zero mean Gaussian random variables with variance $q\Delta t$. The state $|\tilde{\psi}\rangle$ evolves according to

$$d\,|\tilde{\psi}_t\rangle = \left[-\frac{\sigma^2}{2}R_t^\dagger R_t dt + R_t dx_t\right]|\tilde{\psi}_t\rangle\,. \tag{12}$$

Hence our model in Eq. (11) has the form of a non-linear stochastic differential equation.

6.2. Generalization to density matrices

To add expressivity to the model, we can consider starting from a learned density matrix ρ_0, and evolving the density matrix instead of the state $|\tilde{\psi}_t\rangle$. There is the disadvantage that $\rho \in \mathbb{C}^{D\times D}$, so it is more costly to evolve than the pure state. The equation of motion for the (unnormalized) density matrix is

$$\frac{d\tilde{\rho}_t}{dt} = \sigma^2 L(\tilde{\rho}_t) + (\tilde{\rho}_t R_t^\dagger + R_t \tilde{\rho}_t)\frac{dx_t}{dt}, \tag{13}$$

where $L(\cdot)$ is the Linbladian

$$L(\rho) = R_t \rho R_t^\dagger - \frac{1}{2}\left(R_t^\dagger R_t \rho + \rho R_t^\dagger R_t\right). \tag{14}$$

The quantum mechanical average in Eq. (11) then becomes

$$\langle R_t + R_t^\dagger\rangle_t = \frac{\mathrm{Tr}\left[\left(R_t + R_t^\dagger\right)\tilde{\rho}_t\right]}{\mathrm{Tr}\left[\tilde{\rho}_t\right]}. \tag{15}$$

6.3. Parameter estimation

We now have a parametric form of our model and we need to find the values of the parameters that best fit the data, given a dataset. The probability distribution of continuous processes is not normalizable Särkkä and Solin (2019), i.e. if we formally define it as

$$p(\mathcal{X}_t) = \lim_{n\to\infty} p(x_1, ..., x_n), \tag{16}$$

this limit tends to zero or infinity almost everywhere in the domain of the distribution. In order to define a finite loss function, we can consider the relative probability distribution of the process $\mathcal{X}_t$ with respect to the probability of another process that does not contain the learnt parameters. It is natural to define the relative probability of the signal $\mathcal{X}_t$ with respect to the driving Brownian motion β_t. Let us call the probability measure of our model $\mathbb{P}_{\text{cMPS}}(\mathcal{X}_t)$ and the probability distribution associated with Brownian motion $\mathbb{P}_\beta(\mathcal{X}_t)$. According to the Girsanov theorem (Särkkä and Solin (2019)), the relative probability of $\mathbb{P}_{\text{cMPS}}(\mathcal{X}_t)$ with respect to $\mathbb{P}_\beta(\mathcal{X}_t)$ is given by the Radon-Nikodym derivative involved in changing measure from $\mathbb{P}_{\text{cMPS}}(\mathcal{X}_t)$ to $\mathbb{P}_\beta(\mathcal{X}_t)$:

$$\frac{d\mathbb{P}_{\text{cMPS}}(\mathcal{X}_t)}{d\mathbb{P}_\beta(\mathcal{X}_t)} = \exp\left(\frac{A}{q}\int \langle R_t + R_t^\dagger\rangle_t dx_t - \frac{A^2}{2q}\int \langle R_t + R_t^\dagger\rangle_t^2 dt\right), \tag{17}$$

where q is the diffusion constant of the Brownian motion. Our training strategy is to minimise the negative log likelihood (relative to the measure of Brownian motion), i.e., our loss function (associated to a single continuous audio signal $\mathcal{X}_t$) is minus the logarithm of Eq. (17):

$$\text{loss} = -\log\frac{d\mathbb{P}_{\text{cMPS}}(\mathcal{X}_t)}{d\mathbb{P}_\beta(\mathcal{X}_t)} = -A\int \langle R_t + R_t^\dagger\rangle_t dx_t + \frac{A^2}{2}\int \langle R_t + R_t^\dagger\rangle_t^2 dt. \tag{18}$$

Note that we removed the factor $1/q$ because it is not a learning variable.

6.4. Discretization

Since we cannot solve Eq. (11) exactly, we cannot evaluate the likelihood exactly and so we need the aid of discretization methods. We use the Euler-Maruyama integration scheme

$$|\tilde{\psi}_{t+\Delta t}\rangle = \left[\mathbb{1} - \frac{\sigma^2}{2} R_t^\dagger R_t \Delta t + R_t \Delta x_t\right] |\tilde{\psi}_t\rangle\,, \quad \text{and} \tag{19}$$

$$\tilde{\rho}_{t+\Delta t} = \left[\mathbb{1} - \frac{\sigma^2}{2} R_t^\dagger R_t \Delta t + R_t \Delta x_t\right] \tilde{\rho}_t \left[\mathbb{1} - \frac{\sigma^2}{2} R_t^\dagger R_t \Delta t + R_t \Delta x_t\right]^\dagger . \tag{20}$$

The discretization of the model SDE (11) is

$$\Delta x_t = A\langle R_t + R_t^\dagger\rangle_t \Delta t + \Delta\beta_t, \tag{21}$$

where $\Delta x_t \equiv x_{t+\Delta t} - x_t$ and $\Delta\beta_t \equiv \beta_{t+\Delta t} - \beta_t$. The discretization of the loss function (18) is

$$\text{loss}(H, R, A, |\psi_0\rangle) = -A \sum_t \langle R_t + R_t^\dagger\rangle_t \Delta x_t + \frac{A^2}{2} \sum_t \langle R_t + R_t^\dagger\rangle_t^2 \Delta t. \tag{22}$$

Neglecting the constant $-{\Delta x_t}^2/2\Delta t$ and the multiplicative factor $\Delta t/2$, it is expressed as

$$\text{loss}(H, R, A, |\psi_0\rangle) = \sum_t \left(\frac{\Delta x_t}{\Delta t} - A\langle R_t + R_t^\dagger\rangle_t\right)^2 . \tag{23}$$

Note that substituting $\Delta x_t/\Delta t$ by x_t, this is equal to Eq. (8).

6.5. Learnable parameters and hyperparameters

The model is specified by the learnable parameters A, H, R and $|\psi_0\rangle$ (or ρ_0) and the hyperparameters:

1. The bond dimension D, which reflects the complexity of the model.
2. Time discretization Δt. If we set it to be equal to the inverse of the sampling rate of the data, which corresponds to matching the time discretization of the data with the time discretization of the model, we can relate the eigenvalues H to the (angular) frequencies $\omega = 2\pi f$ of the data (see Appendix F).
3. σ, which governs the strength of the term $R^\dagger R$.
4. The hyperparameters σ_ω and σ_R are regularisers for H and R (see Appendix F). In this work we do not experiment with regularisers.

6.6. Sampling

After training, we use the learnt parameters H, R, A and $|\psi_0\rangle$ (or ρ_0) to generate samples using the discrete model

$$x_{t+1} = x_t + \Delta x_t, \text{ where,}$$
$$\Delta x_t = A\langle R_t + R_t^\dagger\rangle_t \Delta t + \sqrt{T}\Delta\beta_t. \tag{24}$$

We introduce a temperature parameter T to tune the variance of the independent increments of the Brownian motion. In generative modeling, it is common to introduce a temperature parameter to optimize the quality of the sampling. See for example Kingma and Dhariwal (2018). At $T = 0$ the generative process is deterministic. As the temperature is increased, the generative model gives rise to a variety of samples due to the randomness of the increments. At very high temperatures, the Gaussian noise dominates the generative process and the samples resemble the training data less and less.

7. Experiments

To test the capabilities of our model, we create synthetic datasets where we know the ground truth probability distributions. We can then readily check whether the learnt probability distribution matches the ground truth. We train on three different datasets: damped sines with random delays, *Gaussian processes* and *filtered Poisson processes*.

7.1. Damped sines

The experimental details are shown in Appendix C.1.

7.1.1. SINGLE FREQUENCY EXPERIMENT

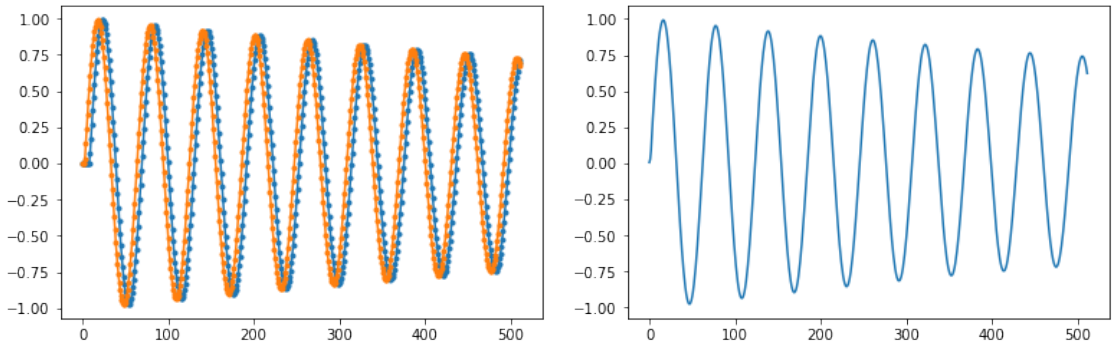
We start by modeling a dataset that consists of damped sines with random delays. Each signal has amplitude zero at the beginning, and the length of this "silence" period is random (see two samples in Fig. 1(a)). All signals have the same frequency $f = 261.6$Hz, the sampling rate is 16KHz and the length of each signal vector is 512 (which corresponds to 0.032 seconds). To generate the training set, we obtain the random delays by sampling from the distribution Gamma$(\alpha = 2, \beta = 0.39)$.

We start by considering the pure state model. The results are:

1. At $T = 0$, the sampling is deterministic given the learned initial state $|\psi_0\rangle$, as explained in Sec. 6.6. The zero temperature sample has the shape of a damped sine with a finite delay. This sample is shown in Fig. 1(b).

2. At finite temperature, we find that we can capture the delay degree of freedom, i.e. different samples have the form of a damped sine, with different delays. On the other hand, we find that the samples have the right form (i.e., the form of a damped sine) for the first 300 points only, having been trained on signals of length 512. In this sense, the outcome of this experiment is not very satisfactory. We experimented with different bond dimensions up to $D = 300$. We show two samples in Fig. 2(a).

We now consider the time evolution of a density matrix

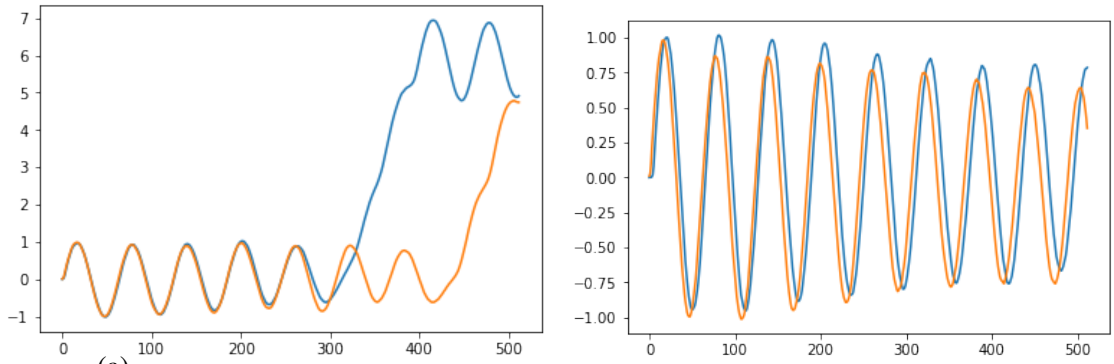


(a) (b)

Figure 1: (a) Two signals of the dataset with different delays. The length of the data samples is 512 and the sampling frequency 16 kHz. (b) The $T = 0$ sample from the our pure state model, after training. It has the form of a damped sine with a finite delay.

$$\tilde{\rho}_{t+\Delta t} = \left[\mathbb{1} - \frac{\sigma^2}{2} R_t^\dagger R_t \Delta t + R_t \Delta x_t\right] \tilde{\rho}_t \left[\mathbb{1} - \frac{\sigma^2}{2} R_t^\dagger R_t \Delta t + R_t \Delta x_t\right]^\dagger . \tag{25}$$

In this case, sampling remains of good quality up to 512 samples, as can be seen in Fig. 2(b).



(a) (b)

Figure 2: (a) Two samples at $T = 30$, $D = 100$ and $\sigma = 10^{-4}$ using the pure state model. The shape of a damped sine is well captured. On the other hand, we can only get proper samples of length 300 approximately. (b) Two samples at $T = 42$, $D = 100$ and $\sigma = 10^{-4}$, where we use a density matrix. Unlike in the pure state case, the samples look like damped sines, for the whole length of 512 samples.

We also experiment with damped sines of two different frequencies. We find that the model learns the manifold of damped sines fairly well, but it fails to capture the two frequencies degree of freedom of the dataset. Details about this experiment are shown in Appendix E.

7.2. Gaussian processes

In the previous section, we tested the ability of our model to learn damped sines. On the other hand, real life sound is a lot more complex than sine waves. For example, real sound is made of several harmonics (unlike a sine wave). To test the capabilities of our model on more realistic data, we move on to training on *Gaussian processes*, specifically *Matérn spectral mixtures* (see Appendix D). The experimental details are shown in Appendix C.2.

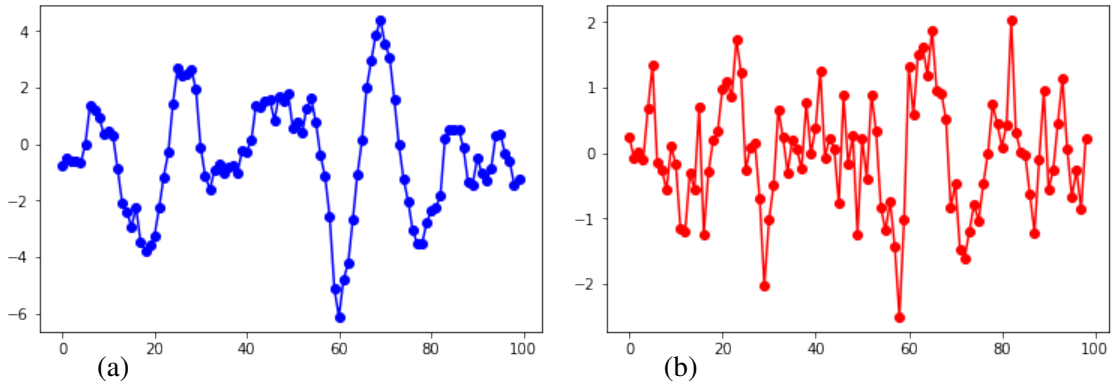


Figure 3: (a) A sample x_t of a Gaussian process with *Matérn spectral mixture* spectral function defined by $(\sigma, \lambda, \omega_0) = (2, 50, 300)$. (b) The increments $\Delta x_t = x_{t+dt} - x_t$ of the sample shown in (a).

We create a dataset of samples of a stationary Gaussian process of choice. We generate the data using a discrete stochastic equation (see Appendix. G.1) instead of sampling from a multivariate Gaussian distribution. We train on two different *Matérn spectral mixture* processes. In the first, the spectral function consists of a single pair of Lorentzians centered at $\omega_0 = \pm 300$ and $(\sigma, \lambda) = (2, 50)$. In the second, we consider a mixture of three frequencies. The mixture is defined by the parameters $(\sigma_i, \lambda_i) = (2, 50)$ for $i = 1, 2, 3$ and $(\omega_1, \omega_2, \omega_3) = (300, 500, 700)$. We show two samples from each of the two datasets in Fig. 4.

7.2.1. RESULTS

Due to the higher complexity of the data compared to the damped sines in Sec. 7.1, instead of just looking at plots of samples, we judge whether the model is successful at learning the above process by 1) calculating the experimental covariance from N samples

$$C_{\text{exp}}(t, t') \equiv \frac{1}{N} \sum_{i=1}^{N} x_i(t) x_i(t'), \tag{26}$$

and comparing it with the exact covariance and 2) checking that the experimental covariance is stationary, i.e. $C_{\text{exp}}(t, t') = C_{\text{exp}}(\tau)$. We find that the model is successful at learning this process and we show the results on Fig. 5. On the other hand, as explained in Sec. 6.6, sampling depends on temperature. The experimental covariance only matches the exact covariance at a given temperature. As one departs from this temperature, the two covariances start to differ. Similarly, the samples are

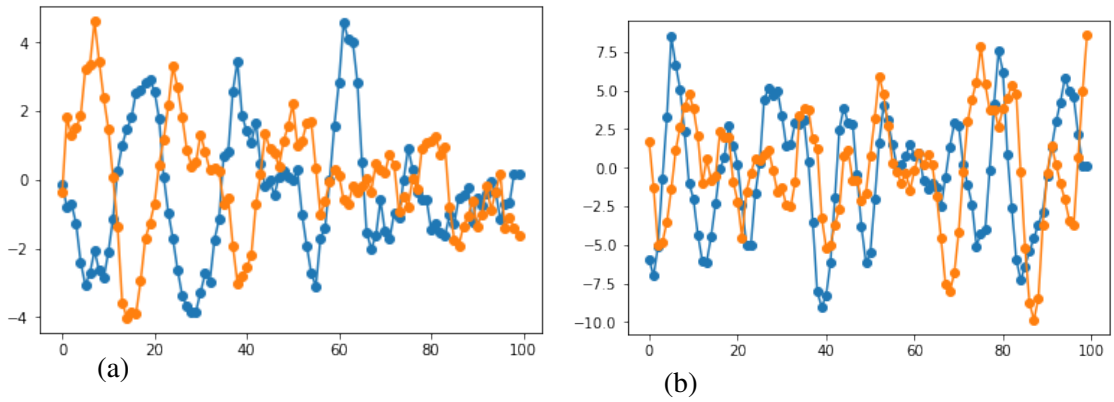


Figure 4: Samples from the two dataset we train on. Two Gaussian processes with *Matérn spectral mixture* spectral function defined by (a) $(\sigma', \lambda, \omega_0) = (2, 50, 300)$ and (b) $(\sigma'_i, \lambda_i) = (2, 50)$ for $i = 1, 2, 3$ and $(\omega_1, \omega_2, \omega_3) = (300, 500, 700)$.

stationary only in a small range of temperatures around this temperature. Furthermore, we find that the experimental covariance becomes stationary only after a few steps, not from the beginning.

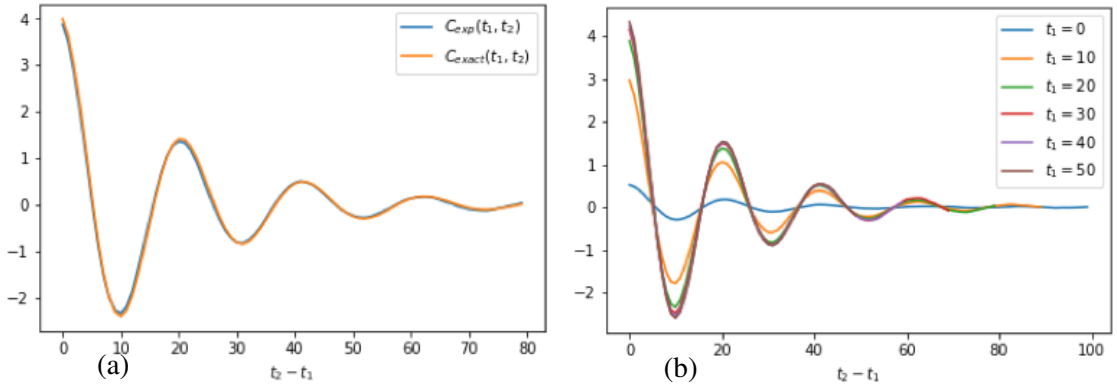


Figure 5: The time indicated in the horizontal axis is an integer index that specifies the time step of the discrete covariance. (a) Perfect match of experimental (blue) and exact (yellow) covariances. The exact covariance has parameters $(\sigma', \lambda, \omega) = (2, 50, 300)$. The experimental covariance is calculated using 40000 samples at $T = 0.00051$. The hyperparameters used are $D = 50$, $dt = 0.001$ and $\sigma = 1$. (b) Experimental covariance $\mathrm{C}_{\mathrm{exp}}(t_1, t_2)$ for different initial times, showing stationarity. It reaches stationarity at $t_1 \approx 20$.

When trained on a mixture of three frequencies, the model succeeds at reproducing stationary samples (after a given time) with the right covariance function. The results are shown in Fig. (6).

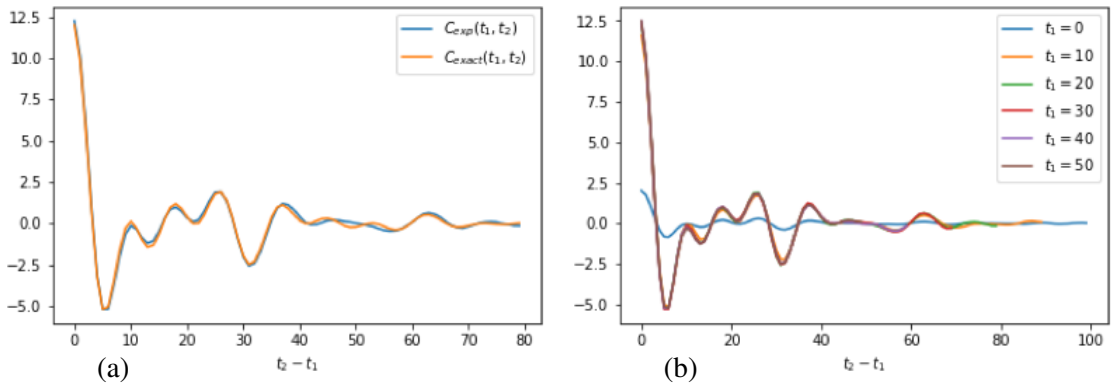


Figure 6: The time indicated in the horizontal axis is an integer index that specifies the time step of the discrete covariance. (a) Perfect match of experimental (blue) and exact (yellow) covariances. The exact covariance has parameters $(\sigma_i', \lambda_i) = (2, 50)$ for $i = 1, 2, 3$ and $(\omega_1, \omega_2, \omega_3) = (300, 500, 700)$. The experimental covariance is calculated using 40000 samples at $T = 0.002$. The hyperparameters used are $D = 100$ and $dt = 0.001$. (b) Experimental covariance $\mathrm{C}_{\mathrm{exp}}(t_1, t_2)$ for different initial times, showing stationarity. Before $t_1 = 10$ the experimental covariance is non-stationary.

7.3. Poisson processes

A feature of stationary Gaussian processes is that because the covariance function is symmetric $\mathrm{C}(t, t') = \mathrm{C}(t', t)$ and all diagonal elements are equal to $\mathrm{C}(t, t)$, the probability density of a given sample $x(t)$ is the same as the probability density of the time-inverted sample. We refer to this symmetry as *time-reversal symmetry* (TRS).

Many real life sounds are not time-reversal symmetric. For example, the chirp of a bird will sound different if played backward. Therefore time-reversal symmetric models like Gaussian models are not suitable to model this kind of sound.

Our cMPS based model is not constrained by time-reversal symmetry, as multivariate Gaussian probability distributions are. We can see this by looking at the discretized time evolution of the unnormalized state. The fact that the one-step time evolution operator does not commute with itself at different times, implies the absence of the TRS constraint.

One can check whether a probability distribution is time-reversal symmetric, from certain correlation functions. Consider the two correlators

$$\begin{aligned}
\mathbb{E}\left[x^3(t_i)x(t_j)\right] &= \int dx(t_1)...dx(t_N)\; x^3(t_i)x(t_j)\; p\left(x(t_1), ..., x(t_i), ..., x(t_j), ..., ..., x(t_N)\right), \\
\mathbb{E}\left[x(t_i)x^3(t_j)\right] &= \int dx(t_1)...dx(t_N)\; x(t_i)x^3(t_j)\; p\left(x(t_1), ..., x(t_i), ..., x(t_j), ..., ..., x(t_N)\right).
\end{aligned} \tag{27}$$

If these two quantities are different, the probability is not invariant under the swap of values of two arguments which implies that it is not TRS.

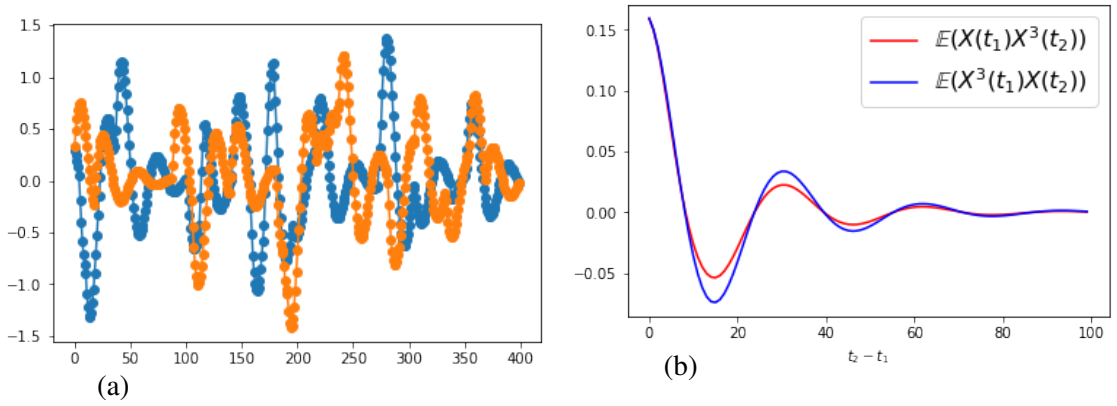


Figure 7: (a) Training samples generated according to the FPP defined in Eq. (28). (b) Exact correlators of FPP defined in Eqs. (29) of the Poisson process defined in Eq. (28). The intensity of the Poisson process is $\lambda = 4$. The amplitude A_k can take values ± 1. The pulse decay time is $\tau = 0.2$ and the angular frequency $\omega = 20$. The time indicated in the horizontal axis is an integer index that specifies the step of the discrete correlator.

We test the ability of our model to learn non-TRS processes, by training it on *filtered Poisson processes*. This process is defined as

$$\begin{aligned} X(t) &= \sum_k A_k \varphi(t - t_k), \text{ where} \\ \varphi(t - t_k) &= \theta(t - t_k) e^{-(t-t_k)/\tau} \sin[\omega(t - t_k)]. \end{aligned} \tag{28}$$

A *filtered Poisson process* (FPP) $X(t)$ consists of a superposition of uncorrelated pulses $\varphi(t - t_k)$, arriving at random times with a Poisson distriution. The overall amplitude A_k is random: at each time, A_k can independently take the values $\pm A$, with equal probabilities. In this process, the correlators defined in Eq. (27) take the form (see the derivation in Appendix H)

$$\begin{aligned} \mathbb{E}\left(X^3(t_1)X(t_2)\right) &= \lambda I_{3,1}^{-\infty,t_1} + 3\lambda^2 I_{1,1}^{-\infty,t_1} I_{2,0}^{-\infty,t_1}, \\ \mathbb{E}\left(X(t_1)X^3(t_2)\right) &= \lambda I_{1,3}^{-\infty,t_1} + 3\lambda^2 I_{1,1}^{-\infty,t_1} \left(I_{0,2}^{-\infty,t_1} + I_{0,2}^{t_1,t_2}\right), \\ I_{n,m}^{t,t'} &= \int_t^{t'} d\alpha\, \varphi^n(t_1 - \alpha)\varphi^m(t_2 - \alpha). \end{aligned} \tag{29}$$

The two correlators are different due to the absence of TRS. We take the initial time $t = -\infty$ because we are interested in the steady state correlators.

The experimental details are shown in C.2. The dataset contains samples of the FPP defined in Eq. (28) with parameters $A = 1$ (i.e., A_k can take values ± 1 at time t_k), $\tau = 0.2$, $\omega = 20$. In order to create steady state signals, we produce signals of length 500, and pick the last 400 points of each signal. This corresponds to signals that have been running for a time 5τ, by when signals are approximately stationary, because the process has a memory time of order τ due to the exponential

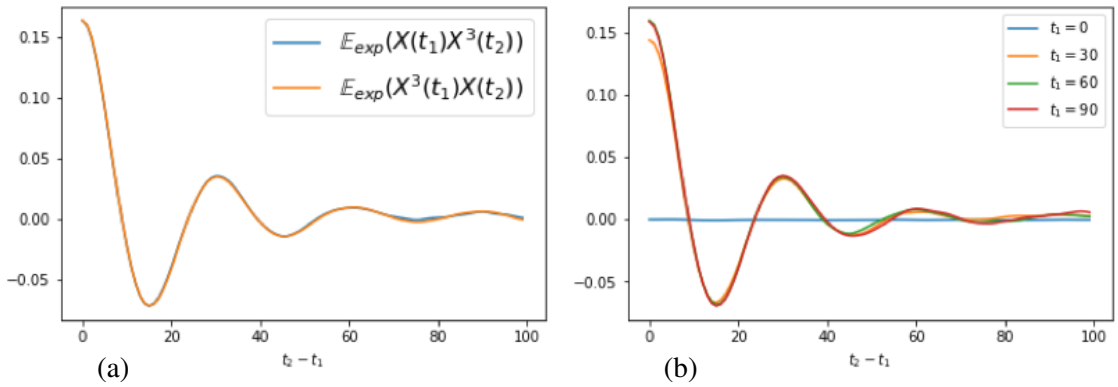


Figure 8: The time indicated in the horizontal axis is an integer index that specifies the time step of the discrete correlator. (a) Experimental correlators $\mathbb{E}_{\text{exp}}\left(X(0)X(t)^3\right)$ and $\mathbb{E}_{\text{exp}}\left(X^3(0)X(t)\right)$. At $T = 0.000012$, they roughly match the exact correlator $\mathbb{E}\left(X^3(t_1)X(t_2)\right)$ shown in Fig. 7(b). The temperature can be tuned to approximately match $\mathbb{E}\left(X(t_1)X^3(t_2)\right)$ instead. But at a given temperature they are both equal, unlike the exact correlators. The hyperparameters used are $dt = 0.01$, $D = 100$ and $\sigma = 1$. The experimental correlators are calculated by averaging over 40000 samples. (b) The experimental correlator $\mathbb{E}_{\text{exp}}\left(X(t_1)X(t_2)^3\right)$, for different values of t_1. It becomes stationary after $t_1 = 30$, approximately.

decay of the pulses $\varphi(t - t_k)$. The Poisson intensity parameter is $\lambda = 4$. We show two steady state training samples in Fig. 7(a).

7.3.1. RESULTS

By tuning the temperature, we can match the experimental correlators to either of the two exact correlators. On the other hand, both experimental correlators we obtain are equal.

8. Conclusions

We introduce a quantum-inspired generative model for raw audio. It is the first machine learning model based on *continuous Matrix Product States*. Our model takes the form of a stochastic Schrödinger equation describing the continuous time measurement of a quantum system. This constitutes a deep autoregressive architecture in which the system's state is a latent representation of past observations.

We rephrase the model in the language of stochastic differential equations. We derive an expression to calculate the two-time characteristic function of a filtered Poisson process. We test our model on three different synthetic datasets. The model is successful at learning single frequency damped sines with random delays but it fails to capture the two frequency degree of freedom. It is able to learn *Matérn spectral mixtures*. Finally, it captures the filtered Poisson process but it fails to discern between $\mathbb{E}\left(X^3(t_1)X(t_2)\right)$ and $\mathbb{E}\left(X(t_1)X^3(t_2)\right)$.

It remains to do a proper hyperparameter tuning considering all the hyperparameters, to see if the performance of the model can be improved. Moreover, and most importantly, the model needs

to be tested on real data: how expressive is the model and how is this related to the bond dimension? How is the quantum entanglement of the model related to the the structure of correlations in the generated samples?

This work opens a new avenue to use matrix product states to model continuous data and we hope that it will set the beginning of the exploration of cMPS for machine learning.

Acknowledgments

We are grateful to Katarzyna Macieszczak for her key insights in the design of the model. BMU and AL would like to acknowledge EPSRC under Grants EP/M506485/1 and EP/P034616/1.

References

Pablo A. Alvarado and Dan Stowell. Efficient learning of harmonic priors for pitch detection in polyphonic music, 2017.

Tai-Danae Bradley, E. Miles Stoudenmire, and John Terilla. Modeling sequences with quantum states: A look under the hood, 2019.

Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 6571–6583. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/7892-neural-ordinary-differential-equations.pdf.

Song Cheng, Lei Wang, Tao Xiang, and Pan Zhang. Tree tensor networks for generative modeling. *Phys. Rev. B*, 99:155131, Apr 2019. doi: 10.1103/PhysRevB.99.155131. URL https://link.aps.org/doi/10.1103/PhysRevB.99.155131.

Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2980–2988. Curran Associates, Inc., 2015. URL http://papers.nips.cc/paper/5653-a-recurrent-latent-variable-model-for-sequential-data.pdf.

Sander Dieleman, Aaron van den Oord, and Karen Simonyan. The challenge of realistic music generation: modelling raw audio at scale. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 8000–8010. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8023-the-challenge-of-realistic-music-generation-modelling-raw-audio-at-scale.pdf.

Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis, 2018.

Stavros Efthymiou, Jack Hidary, and Stefan Leichenauer. Tensornetwork for machine learning, 2019.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Douglas Eck, Karen Simonyan, and Mohammad Norouzi. Neural audio synthesis of musical notes with wavenet autoencoders, 2017.

Glen Evenbly. Number-state preserving tensor networks as classifiers for supervised learning, 2019.

Crispin Gardiner and Peter Zoller. *Quantum World Of Ultra-cold Atoms And Light, Book II: The Physics of Quantum-Optical Devices*. Cold Atoms. Imperial College Press, 2015.

Ivan Glasser, Nicola Pancotti, and J. Ignacio Cirac. Supervised learning with generalized tensor networks, 2018.

Ivan Glasser, Ryan Sweke, Nicola Pancotti, Jens Eisert, and J. Ignacio Cirac. Expressive power of tensor-network factorizations for probabilistic modeling, with applications from hidden markov models to quantum machine learning, 2019.

Chu Guo, Zhanming Jie, Wei Lu, and Dario Poletti. Matrix product operators for sequence-to-sequence learning. *Phys. Rev. E*, 98:042114, Oct 2018. doi: 10.1103/PhysRevE.98.042114. URL https://link.aps.org/doi/10.1103/PhysRevE.98.042114.

Zhao-Yu Han, Jun Wang, Heng Fan, Lei Wang, and Pan Zhang. Unsupervised generative modeling using matrix product states. 2017. doi: 10.1103/PhysRevX.8.031012.

Nal Kalchbrenner, Erich Elsen, Karen Simonyan, Seb Noury, Norman Casagrande, Edward Lockhart, Florian Stimberg, Aaron van den Oord, Sander Dieleman, and Koray Kavukcuoglu. Efficient neural audio synthesis, 2018.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 10215–10224. Curran Associates, Inc., 2018. URL http://papers.nips.cc/paper/8224-glow-generative-flow-with-invertible-1x1-convolutions.pdf.

Zhuan Li and Pan Zhang. Shortcut matrix product states and its applications, 2018.

Ding Liu, Shi-Ju Ran, Peter Wittek, Cheng Peng, Raul Blázquez García, Gang Su, and Maciej Lewenstein. Machine learning by unitary tensor network of hierarchical tree structure. *New Journal of Physics*, 21(7):073059, jul 2019. doi: 10.1088/1367-2630/ab31ef. URL https://doi.org/10.1088%2F1367-2630%2Fab31ef.

Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. Samplernn: An unconditional end-to-end neural audio generation model, 2016.

Alexander Novikov, Mikhail Trofimov, and Ivan Oseledets. Exponential machines, 2016.

Roman Orus. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Annals of Physics*, 349:117 – 158, 2014. ISSN 0003-4916. doi: https://doi.org/10.1016/j.aop.2014.06.013. URL http://www.sciencedirect.com/science/article/pii/S0003491614001596.

Tobias J. Osborne, Jens Eisert, and Frank Verstraete. Holographic quantum states. *Phys. Rev. Lett.*, 105:260401, Dec 2010. doi: 10.1103/PhysRevLett.105.260401. URL https://link.aps.org/doi/10.1103/PhysRevLett.105.260401.

Ryan Prenger, Rafael Valle, and Bryan Catanzaro. Waveglow: A flow-based generative network for speech synthesis, 2018.

Simo Särkkä and Arno Solin. *Applied Stochastic Differential Equations*. Institute of Mathematical Statistics Textbooks. Cambridge University Press, 2019. doi: 10.1017/9781108186735.

C. Schön, E. Solano, F. Verstraete, J. I. Cirac, and M. M. Wolf. Sequential generation of entangled multiqubit states. *Phys. Rev. Lett.*, 95:110503, Sep 2005. doi: 10.1103/PhysRevLett.95.110503. URL https://link.aps.org/doi/10.1103/PhysRevLett.95.110503.

Arno Solin and Simo Särkkä. Explicit link between periodic covariance functions and state space models. In *AISTATS*, 2014.

James Stokes and John Terilla. Probabilistic modeling with matrix product states, 2019.

E Miles Stoudenmire. Learning relevant features of data with multi-scale tensor networks. *Quantum Science and Technology*, 3(3):034003, apr 2018. doi: 10.1088/2058-9565/aaba1a. URL https://doi.org/10.1088%2F2058-9565%2Faaba1a.

Edwin Stoudenmire and David J Schwab. Supervised learning with tensor networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 4799–4807. Curran Associates, Inc., 2016. URL http://papers.nips.cc/paper/6211-supervised-learning-with-tensor-networks.pdf.

Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. Wavenet: A generative model for raw audio, 2016.

F. Verstraete and J. I. Cirac. Continuous matrix product states for quantum fields. *Phys. Rev. Lett.*, 104:190405, May 2010. doi: 10.1103/PhysRevLett.104.190405. URL https://link.aps.org/doi/10.1103/PhysRevLett.104.190405.

Ce Wang, Hui Zhai, and Yi-Zhuang You. Emergent quantum mechanics in an introspective machine learning architecture. 2019. doi: 10.1016/j.scib.2019.07.014.

Wikipedia contributors. Campbell's theorem (probability) — Wikipedia, the free encyclopedia, 2019. URL https://en.wikipedia.org/w/index.php?title=Campbell%27s_theorem_(probability)&oldid=884088554. [Online; accessed 26-June-2019].

William J. Wilkinson, Michael Riis Andersen, Joshua D. Reiss, Dan Stowell, and Arno Solin. Unifying probabilistic models for time-frequency analysis, 2018.

Howard M. Wiseman and Gerard J. Milburn. *Quantum Measurement and Control*. Cambridge University Press, 2009. doi: 10.1017/CBO9780511813948.

Appendix A. Continuous matrix product states

The physical lattice states are well-captured by matrix product states. In the context of continuous quantum systems, there exist a continuum limit, without any reference to an underlying lattice parameter. This family of states are called *continuous matrix product states* (cMPS). They describe field theories in one spatial dimension. Just as MPS captures the entanglement structure of low energy states of quantum spin systems, the entanglement structure of cMPS is tailored to describe the low-energy states of quantum field theories Verstraete and Cirac (2010).

To define a cMPS, let us consider a one-dimensional system of bosons or fermions in a ring of length L. The associated field operators $\hat{\psi}(x)$ obey $[\hat{\psi}(x), \hat{\psi}(y)^\dagger] = \delta(x-y)$ with $0 \leq x, y \leq L$. The cMPS is defined as

$$|\Psi\rangle = \mathrm{Tr}_{aux}\left[\mathcal{P}e^{\int_0^L dx\left[Q(x)\otimes\mathbb{1}+R(x)\otimes\hat{\psi}(x)^\dagger\right]}\right]|0\rangle\,. \tag{30}$$

The matrices $Q(x)$ and $R(x)$ have dimensions $D \times D$ and they act in the D-dimensional auxiliary space. $\mathcal{P}\exp$ is the notation for the path-ordered exponential, Tr_{aux} is a trace over the ancilla and $|0\rangle$ is the vacuum of the field operators. The state becomes translationally invariant when Q and R do not depend on x and a system with open boundary conditions can be obtained by replacing the Tr_{aux} by a left and right multiplication of the auxiliary system with a row and a column vector, respectively:

$$|\Psi\rangle = \langle v_L|\,\mathcal{P}e^{\int_0^L dx\left[Q(x)\otimes\mathbb{1}+R(x)\otimes\hat{\psi}(x)^\dagger\right]}|v_R\rangle\otimes|0\rangle\,. \tag{31}$$

A.1. Connection between MPS and cMPS

As shown by Schön *et al.* in Schön et al. (2005), an MPS with bond dimension D can be seen as a sequentially generated multiqubit state, arising from a D-level system. Let $\mathcal{H}_A = \mathbb{C}^D$ and $\mathcal{H}_B = \mathbb{C}^2$ be the Hilbert spaces of the ancilla and a single qubit respectively. In every step of the sequential generation, we consider unitary evolution of the joint system $\mathcal{H}_A \otimes \mathcal{H}_B$. Assuming that each qubit is initially empty $|0\rangle$, we disregard the qubit at the input, such that the evolution takes the form of an isometry $V:\ \mathcal{H}_A \to \mathcal{H}_A \otimes \mathcal{H}_B$. Choosing a basis in the ancilla space, the isometry is expressed as

$$V = \sum_s \sum_{a,b} A^s_{a,b}\left(|a\rangle\langle b|\otimes|s\rangle\right), \tag{32}$$

where $\sum_s A^{s\dagger}A^s = \mathbb{1}$ is the isometry condition and each A^s is a $D \times D$ matrix. After applying V n times to an initial state $|\psi_\mathrm{I}\rangle \in \mathcal{H}_A$,

$$|\Psi\rangle = \sum_{\mathbf{s}} \sum_{ab} A^{s_n}_{a,}\ldots A^{s_1}_{,b}\,\langle b|\psi_\mathrm{I}\rangle\left(|a\rangle\otimes|\mathbf{s}\rangle\right). \tag{33}$$

The generated n qubits are in general entangled both with the ancilla and between themselves. If the ancilla is decoupled in the last step, the final state is an MPS in the space of the n qubits:

$$|\Psi\rangle = |\psi_\mathrm{F}\rangle\otimes\sum_{\mathbf{s}}\sum_{ab}\langle\psi_\mathrm{F}|a\rangle\, A^{s_n}_{a,}\ldots A^{s_1}_{,b}\,\langle b|\psi_\mathrm{I}\rangle\,|\mathbf{s}\rangle\,. \tag{34}$$

This result shows that all sequentially generated multiqubit states, arising from a D-dimensional ancillary system $\mathcal{H}_A$, are instances of MPS with $D \times D$ matrices A^s and open boundary conditions specified by $|\psi_{\mathrm{I}}\rangle$ and $|\psi_{\mathrm{F}}\rangle$.

Let us now consider the cMPS shown in Eq. (31), without projecting the ancilla onto $|v_L\rangle$. Taking $L = dx$ and Q, R translationally invariant,

$$\begin{aligned}
|\Psi\rangle &= \mathcal{P} e^{dx\left[Q\otimes\mathbb{1}+R\otimes\hat{\psi}(x)^\dagger\right]} |v_R\rangle \otimes |0\rangle \\
&= \left[\mathbb{1}\otimes\mathbb{1} + Q dx \otimes \mathbb{1} + R dx \otimes \hat{\psi}(x)^\dagger\right] |v_R\rangle \otimes |0\rangle \\
&= \sum_{ab} \left[(\delta_{ab} + Q_{ab} dx)\, |a\rangle\, |b\rangle \otimes \mathbb{1} + R_{ab} dx\, |a\rangle\, |b\rangle \otimes \hat{\psi}(x)^\dagger\right] |v_R\rangle \otimes |0\rangle \\
&= \sum_s \sum_{ab} A^s_{ab} \left(|a\rangle\langle b| \otimes |s\rangle\right) |v_R\rangle ,
\end{aligned} \tag{35}$$

where $A^0_{ab} = \delta_{ab} + Q_{ab} dx$, $A^1_{ab} = R_{ab} dx$, ${\psi^\dagger}^0(x) = \mathbb{1}$ and ${\psi^\dagger}^s(x)\,|0\rangle = |s\rangle$. This is just the isometry shown in Eq. (32).

Appendix B. Physical picture of cMPS

In the following we will see that a state of the form cMPS appears in the interaction picture time evolution of a composite state of a D-level system (which we refer to as the *ancilla*), coupled to a quantum field bath. In particular we consider a D-level atom coupled to an electromagnetic field in the dipole approximation. The Hamiltonian of the composite system is

$$H = H_a + H_b + V, \quad \text{where}$$

$$H_a = \sum_n \varepsilon_n |n\rangle\langle n| , \tag{36}$$

$$H_b = \sum_k \omega_k b_k^\dagger b_k, \tag{37}$$

$$V = Ep = \sum_k \left(g_k b_k + g_k b_k^\dagger\right) \sum_{nm} p_{nm} |n\rangle\langle m| . \tag{38}$$

Here, $\{|n\rangle\}$ are the D eigenstates of the atom, $\{b_k\}$ are bosonic annihilation operators for each electromagnetic mode k (the quantum number k contains all the information specifying the mode), and $\{p_{nm}\}$ are the matrix elements of the dipole moment of the atom between different eigenstates. The coefficient g_k can be assumed to be real without loss of generality and it depends on details of the electromagnetic mode k, specifically the volume of the space that the modes occupy Gardiner and Zoller (2015).

For the sake of simplicity, we will consider the case where the atom is a two-level system with energy gap Δ, and so, calling the matrix element between the two levels $p_{10} \equiv p$,

$$V = \sum_k \left(g_k b_k + g_k b_k^\dagger\right) \left(p\, |1\rangle\langle 0| + p^*\, |0\rangle\langle 1|\right). \tag{39}$$

As a first step, we go to the interaction frame with respect to H_b

$$\left|\Psi^i\right\rangle = U_0^\dagger\, |\Psi\rangle , \quad U_0 = e^{-iH_b t}. \tag{40}$$

The corresponding Schrödinger equation in the interaction picture is

$$\partial_t \left|\Psi^i\right\rangle = -i \left(V_{IF} + H_a\right) \left|\Psi^i\right\rangle, \tag{41}$$

where the coupling in the interaction frame takes the form

$$V_{IF} = U_0^\dagger V U_0 = \sum_k \left(g_k b_k e^{-i\omega_k t} + g_k b_k^\dagger e^{i\omega_k t}\right) \left(p \left|1\right\rangle\!\left\langle 0\right| + p^* \left|0\right\rangle\!\left\langle 1\right|\right) \tag{42}$$

$$= \sum_k \left(g_k b_k e^{-i\omega_k t} + g_k b_k^\dagger e^{i\omega_k t}\right) \left(e^{i\Delta t} \frac{p \left|1\right\rangle\!\left\langle 0\right|}{e^{i\Delta t}} + e^{-i\Delta t} \frac{p^* \left|0\right\rangle\!\left\langle 1\right|}{e^{-i\Delta t}}\right). \tag{43}$$

Here, we introduced $1 = e^{i\Delta t}/e^{i\Delta t}$ to be able to perform a *rotating wave approximation* (RWA):

$$V_{IF}^{RWA} = \sum_k \left(g_k b_k^\dagger e^{-i\delta_k t} p^* \left|0\right\rangle\!\left\langle 1\right| e^{i\Delta t} + g_k b_k e^{i\delta_k t} p \left|1\right\rangle\!\left\langle 0\right| e^{-i\Delta t}\right), \tag{44}$$

where $\delta_k \equiv \Delta - \omega_k$ is the detuning. We define the bath operator $b(t) \equiv e^{-i\Delta t} \sum_k g_k b_k e^{i\delta_k t}$ and raising operator $R^\dagger \equiv ip \left|1\right\rangle\!\left\langle 0\right|$. The resulting Schrödinger equation takes the form

$$\partial_t \left|\Psi^i\right\rangle = \left(R b^\dagger(t) - R^\dagger b(t) - i H_a\right) \left|\Psi^i\right\rangle. \tag{45}$$

The time dependence of $b(t)$ stems not only from the fact that we are in the interaction frame but also the $e^{i\Delta t}$ we introduced to perform the RWA. This is equivalently an atomic system in the Schrödinger picture, where the system is driven by these fields, which are regarded as known time-dependent operators Gardiner and Zoller (2015).

These new operators do not follow bosonic commutation relations, instead

$$[b(t), b^\dagger(t')] = e^{-i\Delta(t-t')} \sum_k g_k^2 e^{i\delta_k (t-t')}. \tag{46}$$

For certain baths, $\sum_k g_k^2 e^{i\delta_k (t-t')}$ is sharply peaked at $t = t'$ Wiseman and Milburn (2009). Therefore, we will approximate this function with a delta function:

$$[b(t), b^\dagger(t')] = e^{-i\Delta(t-t')} \delta(t-t') = \delta(t-t'). \tag{47}$$

This corresponds to taking the *Markovian* limit Wiseman and Milburn (2009). In the remainder of the derivation, we define the differential bath operator $dB_t \equiv b(t) dt$. We note that

$$[dB_t, dB_t^\dagger] = \underbrace{\delta(0) dt}_{1} dt = dt. \tag{48}$$

and so $dB_t \sim \sqrt{dt}$. This can be understood by thinking of dt as the smallest unit into which time can be devided. Then we have a discrete delta function with finite width and height, with area $\delta(0) dt = 1$.

We now consider the case where the electromagnetic field is in the vacuum state $\left|0\right\rangle$. Considering a differential step and expanding to order dt

$$\left|\Psi_{dt}^i\right\rangle = \exp\left(R dB_{dt}^\dagger - R^\dagger dB_{dt} - i H_a dt\right) \left|\psi_0^i\right\rangle \otimes \left|0\right\rangle$$

$$\approx \left(\mathbb{1} - \left(i H_a + \frac{R^\dagger R}{2}\right) dt + R dB_{dt}^\dagger + \frac{R^2}{2} dB_{dt}^\dagger dB_{dt}^\dagger\right) \left|\psi_0^i\right\rangle \otimes \left|0\right\rangle. \tag{49}$$

Note that $dB_{dt}\left|0\right\rangle = 0$ and $dB_{dt}dB_{dt}^{\dagger}\left|0\right\rangle = dt\left|0\right\rangle$ from Eq. (48). Neglecting the last term, this is the first order expansion of the continuous matrix product state defined in Eq. (31), given we identify $Q = -iH_a - R^{\dagger}R/2$ and $\hat{\psi}^{\dagger}dt = dB^{\dagger}$. A more careful analysis (beyond the scope of this work) reveals that the last term $dB^{\dagger}dB^{\dagger}$ need not be kept Gardiner and Zoller (2015). Thus we are finally left with a *continuous matrix product state* Osborne et al. (2010); Verstraete and Cirac (2010)

$$\left|\Psi_{dt}^{i}\right\rangle \approx \left[\mathbb{1} - \left(iH_a + \frac{R^{\dagger}R}{2}\right)dt + RdB_{dt}^{\dagger}\right]\left|\psi_0^i\right\rangle \otimes \left|0\right\rangle. \tag{50}$$

As a last step let us consider the time evolution of $\left|\Psi_{dt}^{i'}\right\rangle = e^{iH_a t}\left|\Psi_{dt}^{i}\right\rangle$ so that the model takes a more compact form. Then,

$$\left|\Psi_{t+dt}^{i'}\right\rangle = \left[\mathbb{1} - \frac{R_t^{\dagger}R_t}{2}dt + R_t dB_{t+dt}^{\dagger}\right]\left|\psi_t^{i'}\right\rangle \otimes \left|0\right\rangle, \tag{51}$$

where $R_t = e^{iH_a t}Re^{-iH_a t}$. In the remainder, we will not keep the i' index but we will still be referring to states whose time evolution is (51). Also we will use H to refer to H_a.

B.1. Balanced homodyne detection

Balanced homodyne measurement corresponds to mixing of the output field with a strong (classical) oscillator (mode a) on a balanced beam splitter, and measuring the photon number difference between the two output fields $c = (a+b)/\sqrt{2}$ and $d = (a-b)/\sqrt{2}$:

$$\Delta n = c^{\dagger}c - d^{\dagger}d = ab^{\dagger} + a^{\dagger}b \approx \alpha b^{\dagger} + \alpha^{*}b, \tag{52}$$

where the last approximation follows from the operator Δn acting on the coherent state $|\alpha\rangle$, i.e., $a|\alpha\rangle \approx \alpha|\alpha\rangle$. In particular, the approximation becomes exact for the photon count divided by the oscillator amplitude in the strong oscillator limit,

$$I \equiv \lim_{|\alpha|\to\infty}\frac{c^{\dagger}c - d^{\dagger}d}{|\alpha|} = e^{i\phi}b^{\dagger} + e^{-i\phi}b. \tag{53}$$

We now discuss the effect of the operator $I = e^{i\phi}b^{\dagger} + e^{-i\phi}b$ being measured continuously on the output of an open quantum system described by a cMPS. As shown in Eq. (51),

$$\begin{aligned}\left|\Psi_{t+dt}\right\rangle &= \left[\mathbb{1} - \frac{1}{2}R_t^{\dagger}R_t dt + R_t \otimes dB_{t+dt}^{\dagger}\right]\left|\psi_t\right\rangle \otimes \left|0\right\rangle \\ &= \left[\mathbb{1} - \frac{1}{2}R_t^{\dagger}R_t dt + R_t \otimes (dB_{t+dt}^{\dagger} + e^{-i2\phi}dB_{t+dt})\right]\left|\psi_t\right\rangle \otimes \left|0\right\rangle,\end{aligned} \tag{54}$$

where we introduced $e^{-i2\phi}dB_{t+dt}$ so that we can introduce the operator I in the equation of motion. If we make a measurement of I at time $t+dt$, projecting the state $|\Psi_{t+dt}\rangle$ onto $|I_{t+dt}\rangle \otimes \langle I_{t+dt}|\Psi_{t+dt}\rangle$, we are left with the following state of the ancilla

$$\langle I_{t+dt}|\Psi_{t+dt}\rangle \equiv |\widetilde{\psi}_{t+dt}\rangle = \left[\mathbb{1} - \frac{1}{2}R_t^\dagger R_t dt + R_t\, e^{-i\phi} I_{t+dt} dt\right] |\psi_t\rangle \times \sqrt{\mathcal{P}(I_{t+dt})}, \tag{55}$$

where $\mathcal{P}(I_{t+dt}) = |\,\langle 0|I_{t+dt}\rangle\,|^2 = \sqrt{dt/2\pi}\exp(-dtI_{t+dt}^2/2)$ is the probability of measuring I_{t+dt} on the vacuum state and

$$\begin{aligned} p(I_{t+dt}) = \langle\widetilde{\psi}_{t+dt}|\widetilde{\psi}_{t+dt}\rangle \quad &= \quad \mathcal{P}(I_{t+dt})\Big\{1 + \langle e^{-i\phi}R_t + e^{i\phi}R_t^\dagger\rangle_{\psi_t} I_{t+dt}dt \\ &\quad + \left[-1 + \left(I_{t+dt}\sqrt{dt}\right)^2\right]\langle R_t^\dagger R_t\rangle_{\psi_t}\, dt + \mathcal{O}(I_{t+dt}dt^2, dt^2)\Big\} \end{aligned} \tag{56}$$

is the probability density of obtaining I_{t+dt}. Recalling Eqs. 48 and 53, note that $I \sim 1/\sqrt{dt}$. Then $\left[-1 + \left(I_{t+dt}\sqrt{dt}\right)^2\right]$ is of order one and

$$p(I_{t+dt}) = \sqrt{\frac{dt}{2\pi}}\exp\left[-\frac{dt}{2}I_{t+dt}^2 + \langle e^{-i\phi}R_t + e^{i\phi}R_t^\dagger\rangle_{\psi_t} I_{t+dt}dt + \mathcal{O}(dt)\right]. \tag{57}$$

We now add a term of order dt to complete the square so that we get a Gaussian probability density

$$\begin{aligned} p(I_{t+dt}) &= \sqrt{\frac{dt}{2\pi}}\exp\left[-\frac{dt}{2}\left(I_{t+dt} - \langle e^{-i\phi}R_t + e^{i\phi}R_t^\dagger\rangle_{\psi_t}\right)^2 + \mathcal{O}(dt)\right] \\ &\approx \sqrt{\frac{dt}{2\pi}}\exp\left[-\frac{dt}{2}\left(I_{t+dt} - \langle e^{-i\phi}R_t + e^{i\phi}R_t^\dagger\rangle_{\psi_t}\right)^2\right] \\ &= \sqrt{\frac{1}{2\pi\left(1/\sqrt{dt}\right)^2}}\exp\left[-\frac{\left(I_{t+dt} - \langle e^{-i\phi}R_t + e^{i\phi}R_t^\dagger\rangle_{\psi_t}\right)^2}{2\left(1/\sqrt{dt}\right)^2}\right]. \end{aligned} \tag{58}$$

Equivalently,

$$I_{t+dt} = \langle e^{-i\phi}R_t + e^{i\phi}R_t^\dagger\rangle_{\psi_t} + z, \text{ where } z \sim N(0, 1/dt), \tag{59}$$

where $N(0, 1/dt)$ is a Gaussian distribution with zero mean variance $1/dt$. For the remainder of the paper we fix $\phi = 0$. The conditional joint probability density for a sequence of measurements $\{I_t\}$ is

$$\begin{aligned} p(I_T, ..., I_1|H, R) &= \prod_{k=0}^{T-1} p(I_{k+1}|I_k, ..., I_1; H, R), \text{ where} \\ p(I_{k+1}|I_k, ..., I_1; H, R) &= \sqrt{\frac{1}{2\pi\left(1/\sqrt{dt}\right)^2}}\exp\left[-\frac{\left(I_{k+1} - \langle R_t + R_t^\dagger\rangle_{\psi_k}\right)^2}{2\left(1/\sqrt{dt}\right)^2}\right], \end{aligned} \tag{60}$$

where ψ_k is the state of the ancilla at time k.

Aside from a few last details and refinements that are described in the main text, this probability distribution defines our *quantum-inspired model*.

Appendix C. Experimental details

C.1. Damped sines

1. We consider the time derivative of the data as the homodyne current, i.e., $I_t = dx_t/dt$. The matrix R is complex and we set its diagonal elements to zero. Hence, we only keep oscillatory parts of $R(t)$, which we consider appropriate to model oscillatory data.

2. We learn the initial state $|\psi_0\rangle$ (or ρ_0). When using density matrices, we parameterize ρ_0 by $\rho_0 = \frac{W^\dagger W}{\mathrm{tr}[W^\dagger W]}$ to enforce normalization and real and positive eigenvalues. The matrix $W \in \mathbb{C}^{r\times D}$ defines the rank of the initial density matrix, with $r = 1$ corresponding to an initial pure state $|\psi_0\rangle$.

3. We use regularisers for the elements of H and R. These are set to $\sigma_\omega^2 = \frac{(16000\pi)^2}{400}$ and $\sigma_R^2 = 5$, defined in Appendix F. They are included in the model as a refinement, but we do not experiment with them. They will become important when training on real data, which is more complex than the data considered here.

4. The hyperparameter dt remains fixed to $dt = 1/16000$. We experiment with different values of D and σ but only show results with the values that give the best results.

5. The batch size is 8.

C.2. GP and FPP

1. We consider the data to be the homodyne current, i.e. $I_t = x_t$. This is because as shown in Fig. 3, on this dataset the increments of the signal are a lot more spiky than the signals themselves, which makes learning difficult in the continuous formulation of the problem.

2. We learn the initial state $|\psi_0\rangle$.

3. The hyperparameter dt is set to $dt = 0.001$ and $\sigma = 1$. We experiment with different values of D and σ but only show results with the values that give the best results.

4. We do not use regularisers.

5. The batch size is 8.

Appendix D. Details about Gaussian Processes

A stochastic function $x(t)$ is a Gaussian process (GP) if any finite collection of random variables $x(t_1), ..., x(t_n)$ have a multidimensional Gaussian distribution Särkkä and Solin (2019). A GP is defined in terms of its *mean* $\mathrm{m}(t)$ and its *covariance function* (or *kernel*) $\mathrm{C}(t, t')$, defined as

$$\mathrm{m}(t) = \mathbb{E}[x(t)], \tag{61}$$

$$\mathrm{C}(t, t') = \mathbb{E}\left[(x(t) - \mathrm{m}(t))\left(x(t') - \mathrm{m}(t')\right)\right]. \tag{62}$$

A Gaussian process is *stationary* if the mean is time independent and the covariance function only depends on time differences

$$\mathrm{C}(t, t') = \mathrm{C}(t - t'). \tag{63}$$

We use the notation $\mathrm{C}(\tau)$ (where $\tau \equiv t - t'$) when considering stationary processes. The Wiener-Khintchine theorem relates the stationary kernel to a corresponding spectral function

$$S(\omega) = \int_{-\infty}^{\infty} d\tau\, \mathrm{C}(\tau) e^{-i\omega\tau}, \tag{64}$$

$$\mathrm{C}(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} d\omega\, S(\omega) e^{i\omega\tau}. \tag{65}$$

A specific kind of stationary GPs that have been used to reflect the complex harmonic structure of musical notes are *Matérn spectral mixtures*. They have been used for different sound related machine learning tasks Alvarado and Stowell (2017). Consider the kernels

$$\mathrm{C}_{1/2}(\tau) = \sigma^2 e^{-\lambda\tau}, \tag{66}$$

$$\mathrm{C}_{\cos}(\tau) = \cos(\omega_0 \tau). \tag{67}$$

The corresponding spectral densities are

$$S_{1/2}(\omega) = \frac{2\sigma^2\lambda}{\lambda^2 + \omega^2}, \tag{68}$$

$$S_{\cos}(\omega) = \pi \left[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)\right]. \tag{69}$$

The spectral density of the product of the two kernels $C(\tau) = C_{1/2}(\tau) C_{\cos}(\tau)$ takes the form of a pair of Lorentzians centered at $\pm\omega_0$

$$S(\omega; \boldsymbol{\theta}) = 2\pi\sigma^2\lambda \left[\frac{1}{\lambda^2 + (\omega - \omega_0)^2} + \frac{1}{\lambda^2 + (\omega + \omega_0)^2}\right], \tag{70}$$

where $\boldsymbol{\theta} = (\sigma, \lambda, \omega_0)$. The general form of *Matérn spectral mixtures* is a sum over different pairs of Lorentzians

$$S_{\mathrm{SMS}}(\omega; \boldsymbol{\Theta}) = \sum_{j=1}^{N} S(\omega; \boldsymbol{\theta}_j), \tag{71}$$

where $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_j\}$. The corresponding covariance is

$$\mathrm{C}_{\mathrm{MSM}}(\tau; \boldsymbol{\Theta}) = \sum_{j=1}^{N} \sigma_j^2 e^{-\lambda_j \tau} \cos(\omega_j \tau). \tag{72}$$

Appendix E. Two frequencies experiment

We want to see if we can generate samples of two different frequencies, after training on a dataset of damped sines with random delays and two different frequencies $f = 600, 800$Hz. The length of the training sequences is 100 samples.

We start with the pure state model. We train the model on a dataset that only contains two signals, shown in Fig. 9(a). After training, our model generates signals with different frequencies that lie in between the two frequencies of the dataset, as shown in Fig. 9(b). The frequencies of the samples seem to be closer to $f = 800$Hz than to $f = 600$Hz.

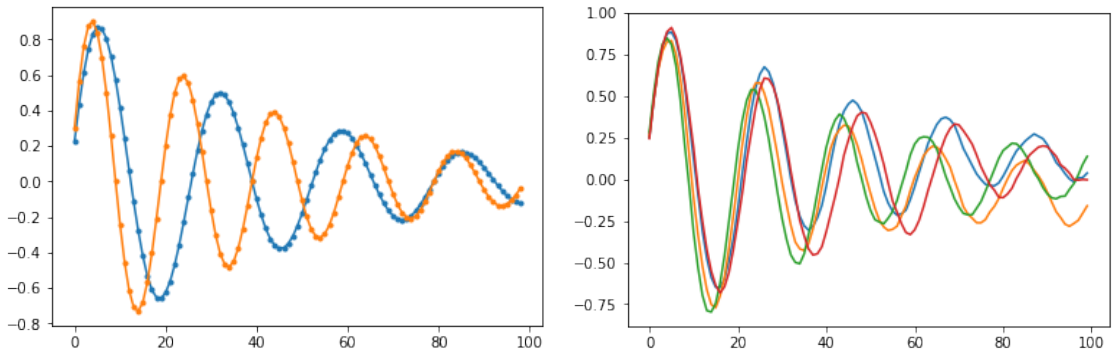


Figure 9: (a) Training set made of two signals. (b) Four samples at T=100 and $D = 50$ and $\sigma = 10^{-4}$, after having trained on the two signal dataset shown in (a).

We then move on to modeling a dataset of damped sines with random delays like we did in Sec. 7.1.1, but this time the training set will contain damped sines of two different frequencies as shown in Fig.10(a). We find that after training, samples are always quite close to the higher frequency. Different generated samples have different shapes, but all look like damped sines. The model learns the manifold of damped sines fairly well, but it fails to capture the two frequencies degree of freedom of the dataset, in that there are no samples with frequencies close to $f = 600$Hz. We show the result in Fig. 10(b).

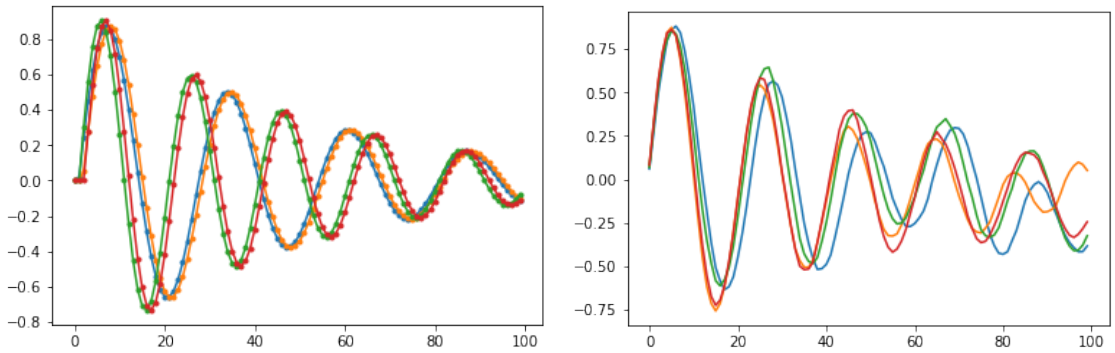
We experiment considering the time evolution of a density matrix but the performance of the model does not improve compared to the pure state case, i.e. it fails to capture the two frequencies degree of freedom of the dataset.

Appendix F. Regularization

What should be the range of values of the learnt parameters? If we had any intuition or knowledge about this question, we could use it to bias the learning. The way to bias or constrain learning is to introduce regularizers. This is equivalent to introducing a prior and doing maximum a posteriori instead of maximum likelihood.

Regularization of H

Consider the following discretized Scrödinger equation in the interaction picture



(a) (b)

Figure 10: (a) Four signals of the dataset with different delays and frequencies $f = 600, 800$ Hz. The length of the data samples is 100 and the sampling frequency 16 kHz. (b) Four samples at $T = 120, D = 100$ and $\sigma = 10^{-4}$.

$$|\tilde{\psi}_{t+\Delta t}\rangle = \left[\mathbb{1} - \frac{\sigma^2}{2} R^\dagger(t)R(t)\Delta t + R(t)\Delta x_t\right] |\tilde{\psi}_t\rangle \tag{73}$$

where $R(t) \equiv e^{iHt} R e^{-iHt}$. If H is diagonal with eigenvalues ω_n, the matrix elements of R are $R_{ab}(t) = R_{ab} e^{i(\omega_a - \omega_b)t}$. Suppose we want to learn a single sequence, e.g., $x_t = \sin(\omega t)$ [i.e., $\Delta x_t / \Delta t \approx \omega \cos(\omega t)$], such that the loss function is

$$\text{loss} = \sum_t \Big(\omega \cos(\omega t) - A\langle R(t) + R^\dagger(t)\rangle_t\Big)^2. \tag{74}$$

The loss function is minimised when $A\langle R(t)+R^\dagger(t)\rangle_t = \omega\cos(\omega t)$, i.e. when the expectation value $\langle R(t) + R^\dagger(t)\rangle_t$ oscillates with frequency ω. Since matrix elements of both $R(t)$ and $R^\dagger(t)R(t)$ oscillate with frequencies that are differences of eigenvalues of H, it is intuitive that the learned diagonal elements of H should be related to the frequency ω of the training data.

If we assume that H is related to the frequencies, it makes sense to limit it to the bandwidth of audio. *Nyquist's theorem* states that in order to correctly capture a discrete signal, the sampling rate must be at least double the highest frequency contained in the signal. Conversely, the highest frequency that can be captured at a given sampling rate is half the sampling frequency. This frequency is called the *Nyquist frequency*. If differences of eigenvalues of H give frequencies, the spectrum of H should be limited to $\pm s/4$, where s is the sampling rate. Thus if we set the standard deviation of the frequencies to be $\sigma_f = s/4$, and bearing in mind that $\omega = 2\pi f$, the regularization term in the loss should be

$$\mathcal{L}_H = \frac{1}{2\sigma_\omega^2} \sum_n \omega_n^2 = \frac{1}{8\pi^2\sigma_f^2} \sum_n \omega_n^2 = \frac{2}{\pi^2 s^2} \sum_n \omega_n^2, \tag{75}$$

which, up to a constant, corresponds to the logarithm of the Gaussian prior

$$p(\omega_1, ..., \omega_D) = \prod_{n=1}^{D} \sqrt{\frac{1}{2\pi\sigma_\omega^2}} \exp\left(-\frac{\omega_n^2}{2\sigma_\omega^2}\right). \tag{76}$$

Regularization of R

The scale of the signal is set by $A\langle R + R^\dagger\rangle_t$. If the typical scale of the matrix elements of R is r, its value should be determined by $\Delta x = Ar\Delta t$. If we set $A = 1$, we could introduce a Gaussian prior so that $\sigma_R = \Delta x/\Delta t$. The hyperparameter Δx can be inferred from the data. Then,

$$\mathcal{L}_R = \frac{1}{2\sigma_R^2} \sum_{ij} |r_{ij}|^2. \tag{77}$$

On the other hand, in general A is a learning variable and so it is not obvious what the regularizer of R should be.

Appendix G. Relation between covariance functions and SDEs

Most of the theory explained in this appendix can be found in Chapters 6 and 12 of Särkkä and Solin (2019). A stochastic differential equation (SDE) is a differential equation that contains terms which are random functions. This implies that their solutions are also random functions. Consider a Gaussian noise-driven ordinary differential equation of the form

$$d\mathbf{x} = \mathbf{f}(\mathbf{x}, t)dt + \mathbf{L}(\mathbf{x}, t)d\boldsymbol{\beta}(t), \tag{78}$$

where $\boldsymbol{\beta}(t)$ is Brownian motion with diffusion matrix $\mathbf{Q}$ and $\mathbf{f}(\mathbf{x}, t)$ and $\mathbf{L}(\mathbf{x}, t)$ are arbitrary vector- and matrix-valued functions, respectively. The solutions $\mathbf{x}(t)$ of SDEs are random processes and therefore they have certain probability distribution $p(\mathbf{x}(t))$ [also denoted $p(\mathbf{x}, t)$]. This probability density solves the *Fokker–Planck–Kolmogorov* (FPK) equation

$$\begin{aligned}\frac{\partial p(\mathbf{x}, t)}{\partial t} = &- \sum_i \frac{\partial}{\partial x_i} \left[f_i(\mathbf{x}, t)p(\mathbf{x}, t)\right] \\ &+ \frac{1}{2} \sum_{i,j} \frac{\partial^2}{\partial x_i \partial x_j} \left\{\left[\mathbf{L}(\mathbf{x}, t)\mathbf{Q}\mathbf{L}^T(\mathbf{x}, t)\right]_{ij} p(\mathbf{x}, t)\right\},\end{aligned} \tag{79}$$

given the initial condition $p(\mathbf{x}, t_0)$. One can obtain the equations of motion for the mean, covariance and other statistical quantities from this equation. Among others, denoting the mean $\mathbf{m}(t) = \mathbb{E}[\mathbf{x}(t)]$ and the *marginal covariance* $\mathbf{C}(t, t) \equiv \mathbf{P}(t) = \mathbb{E}\left[(\mathbf{x}(t) - \mathbf{m}(t))\,(\mathbf{x}(t) - \mathbf{m}(t))^T\right]$,

$$\frac{d\mathbf{m}}{dt} = \mathbb{E}\left[\mathbf{f}(\mathbf{x}, t)\right], \tag{80}$$

$$\frac{d\mathbf{P}}{dt} = \mathbb{E}\left[\mathbf{f}(\mathbf{x}, t)(\mathbf{x} - \mathbf{m})^T\right] + \mathbb{E}\left[(\mathbf{x} - \mathbf{m})\mathbf{f}^T(\mathbf{x}, t)\right] + \mathbb{E}\left[\mathbf{L}(\mathbf{x}, t)\mathbf{Q}\mathbf{L}^T(\mathbf{x}, t)\right]. \tag{81}$$

Another useful quantity that we can obtain from the FPK equation (79) is the *transition density* $p(\mathbf{x}(t)|\mathbf{x}(s))$ of the SDE in (78), which is the probability of the random process taking the value

$\mathbf{x}(t)$ at time t, given the value at time s was $\mathbf{x}(s)$. This quantity is the solution of the FPK equation (79), with the initial condition $p(\mathbf{x}(t)|\mathbf{x}(s)) = \delta(\mathbf{x}(t) - \mathbf{x}(s))$ at $t = s$.

An SDE is *linear* if $\mathbf{f} = \mathbf{F}\mathbf{x}$. The covariance function $\mathbf{C}(t, t')$ of linear stochastic differential equations can be obtained from the marginal covariance

$$\mathbf{C}(t, t') = \begin{cases} \mathbf{P}(t) \exp\left[(t' - t)\mathbf{F}\right]^T, & \text{if } t < t', \\ \exp\left[(t - t')\mathbf{F}\right] \mathbf{P}(t'), & \text{if } t \geq t'. \end{cases} \tag{82}$$

G.1. Equivalent discretisations of linear time-invariant SDEs

An SDE is *time-invariant* if $\mathbf{f}$ and $\mathbf{L}$ do not depend on time. Consider the linear time-invariant stochastic differential equation

$$d\mathbf{x} = \mathbf{F}\mathbf{x}dt + \mathbf{L}d\boldsymbol{\beta}, \tag{83}$$

with initial conditions $\mathbf{x}(t_0) \sim \mathrm{N}(\mathbf{m}_0, \mathbf{P}_0)$, where $\mathrm{N}(\mathbf{m}_0, \mathbf{P}_0)$ denotes a Gaussian distribution with mean $\mathbf{m}_0$ and marginal covariance $\mathbf{P}_0$. From the FPK equation, one obtains the transition density

$$p\left(\mathbf{x}(t)|\mathbf{x}(s)\right) = N\left(\mathbf{m}(t|s), \mathbf{P}(t|s)\right), \tag{84}$$

where

$$\mathbf{m}(t|s) = \exp\left(\mathbf{F}(t - s)\right) \mathbf{x}(s), \tag{85}$$

$$\mathbf{P}(t|s) = \int_s^t \exp\left(\mathbf{F}(t - \tau)\right) \mathbf{L}\mathbf{Q}\mathbf{L}^T \exp\left(\mathbf{F}(t - \tau)\right)^T d\tau. \tag{86}$$

Let us consider discrete times $\{t_k\}$, separated by Δt_k. Eq. (84) then implies

$$\mathbf{x}(t_{k+1}) - \mathbf{m}(t_{k+1}|t_k) = \mathbf{q}_k, \quad \mathbf{q}_k \sim N(0, \mathbf{P}(t_{k+1}|t_k)). \tag{87}$$

Therefore, we derive a discrete stochastic equation

$$\mathbf{x}(t_{k+1}) = \exp\left(\mathbf{F}(\Delta t_k)\right) \mathbf{x}(t_k) + \mathbf{q}_k, \quad \mathbf{q}_k \sim N(0, \mathbf{P}(\Delta t_k|0)). \tag{88}$$

This discretization is exact in that the probability distribution of the continuous and discrete models defined in Eqs. (78) and (88), coincide at times $\{t_k\}$.

G.2. From steady state covariance functions to discrete stochastic processes

As shown at the beginning of Appendix G, it is possible to derive the covariance function of an SDE. Conversely, it is also possible to find the SDE that corresponds to a given covariance function. Consider the following steady state covariance

$$\begin{aligned} &\mathbf{C}(\tau) = \mathbf{C}_{\cos}(\tau)\mathbf{C}_{\exp}(\tau), \text{ where} \\ &\mathbf{C}_{\cos}(\tau) = \cos(\omega\tau), \\ &\mathbf{C}_{\exp}(\tau) = \sigma^2 e^{-\lambda|\tau|}. \end{aligned} \tag{89}$$

As shown in Wilkinson et al. (2018); Solin and Särkkä (2014), the corresponding SDE is

$$\begin{aligned} d\mathbf{g}(t) &= \mathbf{F}\mathbf{g}(t)dt + \mathbf{L}d\boldsymbol{\beta}, \\ x(t) &= \mathbf{H}\mathbf{g}(t), \end{aligned} \tag{90}$$

where

$$\mathbf{F} = \begin{pmatrix} -\lambda & -\omega \\ \omega & -\lambda \end{pmatrix}, \tag{91}$$

$$\mathbf{L} = \mathbb{1}_2, \tag{92}$$

$$\mathbf{H} = (1, 0). \tag{93}$$

This is called a *continuous state space model*, the vector $\mathbf{H}$ is the *measurement model* and $\mathbf{g}(t)$ the *state*. The equivalent discretization of Eq. (90) is

$$\mathbf{g}_{k+1} = \mathbf{A}\mathbf{g}_k + \mathbf{q}_k, \quad \mathbf{q}_k \sim N(0, \boldsymbol{\Sigma}), \tag{94}$$

$$x(t_k) = \mathbf{H}\mathbf{g}_k, \tag{95}$$

where

$$\mathbf{A} = \exp(-\lambda\Delta t) \begin{pmatrix} \cos(\omega\Delta t) & -\sin(\omega\Delta t) \\ \sin(\omega\Delta t) & \cos(\omega\Delta t) \end{pmatrix}, \tag{96}$$

$$\boldsymbol{\Sigma} = \sigma^2(1 - e^{-2\lambda\Delta t})\mathbb{1}_2. \tag{97}$$

When the kernel is a sum of N stationary kernels $\mathbf{C}(\tau) = \sum_{i=1}^{N} \mathbf{C}_i(\tau)$, we get the corresponding SDE by replacing $\mathbf{F}, \mathbf{L}$ and $\mathbf{H}$ in Eq. (90) by

$$\mathbf{F} = \text{blkdiag}(\mathbf{F}_1, ..., \mathbf{F}_N), \tag{98}$$

$$\mathbf{L} = \text{blkdiag}(\mathbf{L}_1, ..., \mathbf{L}_N), \tag{99}$$

$$\mathbf{Q} = \text{blkdiag}(\mathbf{Q}_1, ..., \mathbf{Q}_N), \tag{100}$$

$$\mathbf{H} = (\mathbf{H}_1, ..., \mathbf{H}_N). \tag{101}$$

Here $\mathbf{Q}$ is the diffusion matrix of Brownian motion $\boldsymbol{\beta}(t)$. The corresponding equivalent discretization is obtained by performing the equivalent substitution of $\mathbf{A}, \boldsymbol{\Sigma}$ and $\mathbf{H}$ in Eq. (94). The dimension of the state vector $\mathbf{g}$ in Eqs. (90) and (94) is then increased by a factor of N.

Obtaining samples from these discrete stochastic processes is equivalent to sampling the corresponding multidimensional Gaussian distributions.

Appendix H. Filtered Poisson processes

H.1. Poisson process

A *standard Poisson process* N_t is a counting process that has jumps of size $+1$ at homogeneously distributed random times and its path is constant in between two jumps. This is defined as

$$N_t = \sum_{k=1}^{\infty} \mathbb{1}_{[t_k,\infty)}, \quad \text{for } t \geq 0, \tag{102}$$

where

$$\mathbb{1}_{[t_k,\infty)} = \begin{cases} 1, & \text{if } t \geq t_k, \\ 0, & \text{if } 0 \leq t < t_k. \end{cases} \tag{103}$$

Furthermore, a Poisson process satisfies the following conditions:

1. Independence of increments: for all $0 \leq t_0 < t_1 < \cdots < t_n$, the increments

$$N_{t_1} - N_{t_0}, ..., N_{t_n} - N_{t_n - 1}, \tag{104}$$

   are independent random variables.

2. Stationarity of increments: $N_{t+h} - N_{s+h}$ and $N_t - N_s$ have the same distribution for all $h > 0$ and $0 \leq s \leq t$.

3. Conditions 1 and 2 imply that the probability distribution of the increments is a Poisson distribution, i.e. for all $0 \leq s \leq t$,

$$p(N_t - N_s = k) = e^{-\lambda(t-s)} \frac{(\lambda(t-s))^k}{k!}. \tag{105}$$

   The parameter λ is called the intensity of the Poisson process.

From the last condition we can infer the sort time asymptotics

$$\begin{aligned} p(N_{\Delta t} = 0) &= e^{-\Delta t \lambda} = 1 - \Delta t \lambda + \mathcal{O}(\Delta t^2) \approx 1 - \Delta t \lambda, \\ p(N_{\Delta t} = 1) &= \Delta t \lambda e^{-\Delta t \lambda} = \Delta t \lambda + \mathcal{O}(\Delta t^2) \approx \Delta t \lambda, \quad \Delta t \to 0. \end{aligned} \tag{106}$$

H.2. One-time characteristic function of a filtered Poisson process

A *filtered Poisson process* (FPP) $X(t)$ consists of the superposition of uncorrelated pulses $\varphi\left(t - t_k\right)$, where the arrival times $\{t_k\}$ follow a Poisson distribution

$$X(t) = \sum_k A_k \varphi\left(t - t_k\right). \tag{107}$$

The overall amplitude A_k is random. Let us consider the case where at each time, A_k can independently take the values $\pm A$, with equal probabilities. The characteristic function of $X(t)$ is

$$\Phi_X(u,t) = \mathbb{E}\left(e^{iuX(t)}\right), \tag{108}$$

where the average is taken over all possible Poisson processes. Note that this involves averaging over the random set of jump times $\{t_k\}$ as well as the value of the sequence of amplitudes $\{A_k\}$ at times $\{t_k\}$. Using Campbell's theorem Wikipedia contributors (2019),

$$\mathbb{E}\left(e^{iuX(t)}\right) = \exp\left\{-\lambda \int_0^t d\alpha \left[1 - \Phi_A(iu\varphi(t-\alpha))\right]\right\}, \text{ where} \tag{109}$$

$$\Phi_A(iu\varphi(t-\alpha)) = \mathbb{E}_A\left(e^{iuA\varphi(t-\alpha)}\right). \tag{110}$$

Naive proof:

Let us rewrite the process as a sum over all time steps $\{i\Delta t\}$, instead of the jump times $\{t_k\}$ only

$$X(t) = \sum_{k\in\{t_k\}} A_k \varphi\left(t - t_k\right) = \sum_{i=1}^{N} \sigma_i A_i \varphi\left(t - i\Delta t\right), \tag{111}$$

where $t = N\Delta t$. The parameter σ_i is 1 if there is a jump, and 0 otherwise. The expectation value in (108) is an average over the random variables A and σ at each time step, i.e.,

$$\Phi_X(u,t) = \mathbb{E}\left(e^{iuX(t)}\right) = \mathbb{E}_\sigma\left[\mathbb{E}_A\left(e^{iuX(t)}\right)\right]. \tag{112}$$

Let us define

$$\Phi_{A_i}\left[u\varphi(t-i\Delta t)\right] \equiv \mathbb{E}_{A_i}\left(e^{iu\sigma_i A_i \varphi(t-i\Delta t)}\right). \tag{113}$$

Then expectation value at time $t_i = i\Delta t$ is

$$\mathbb{E}\left(e^{iu\sigma_i A_i \varphi(t-i\Delta t)}\right) = \mathbb{E}_{\sigma_i}\left\{\Phi_{A_i}\left[u\varphi(t-i\Delta t)\right]\right\}. \tag{114}$$

The probability for there being a jump (i.e., $\sigma = 1$) is $\lambda\Delta t$ and the probability of no jump (i.e., $\sigma = 0$) is $1 - \lambda\Delta t$. Then

$$\mathbb{E}\left(e^{iu\sigma_i A_i \varphi(t-i\Delta t)}\right) = 1 + \lambda\Delta t\left(\Phi_{A_i}\left[u\varphi(t-i\Delta t)\right] - 1\right) \approx \exp\left(-\lambda\Delta t\left\{1 - \Phi_{A_i}\left[u\varphi(t-i\Delta t)\right]\right\}\right). \tag{115}$$

The product over all time steps yields (109). If the amplitude outcomes are $\pm A$ with equal probabilities, then

$$\mathbb{E}\left(e^{iuX(t)}\right) = \exp\left\{-\lambda \int_0^t d\alpha \left[1 - \cos\left(uA\varphi(t-\alpha)\right)\right]\right\}. \tag{116}$$

H.3. Two-time characteristic function

The two-time characteristic function is

$$\Phi_x(u_1,t_1;u_2,t_2)=\mathbb{E}\left(e^{i[u_1X(t_1)+u_2X(t_2)]}\right). \tag{117}$$

Let us consider the case where $t_2>t_1$. Then $X(t_1)$ depends on all the jumps before t_1 and $X(t_2)$ depends on all jumps before t_2, which includes all those that contributed to $X(t_1)$: this is the source of correlation between the two variables. If we split up $X(t_2)$ as

$$X(t_2)=\sum_{\{k:t_1<t_k<t_2\}}A_k\varphi(t_2-t_k)+\sum_{\{k:t_1>t_k\}}A_k\varphi(t_2-t_k), \tag{118}$$

the exponent in Eq. (117) is expressed as the sum of independent quantities

$$u_1X(t_1)+u_2X(t_2)=\sum_{\{k:t_k<t_1\}}A_k\left[u_1\varphi(t_1-t_k)+u_2\varphi(t_2-t_k)\right]+\sum_{\{k:t_1<t_k<t_2\}}u_2A_k\varphi(t_2-t_k). \tag{119}$$

Therefore, the characteristic function becomes

$$\Phi_x(u_1,t_1;u_2,t_2)=\mathbb{E}\left(e^{i\sum_{\{k:t_1>t_k\}}A_k[u_1\varphi(t_1-t_k)+u_2\varphi(t_2-t_k)]}\right)\mathbb{E}\left(e^{i\sum_{\{k:t_1<t_k<t_2\}}u_2A_k\varphi(t_2-t_k)}\right). \tag{120}$$

Following the same procedure we followed to derive the one-time characteristic function, we arrive to

$$\begin{aligned}\Phi_X(u_1,t_1;u_2,t_2)=\exp\Bigg\{&-\lambda\int_{t_1}^{t_2}\left[1-\Phi_A(iu_2\varphi(t_2-\alpha))\right]d\alpha\\&-\lambda\int_0^{t_1}\left[1-\Phi_A(iu_1\varphi(t_1-\alpha)+iu_2\varphi(t_2-\alpha))\right]d\alpha\Bigg\}.\end{aligned} \tag{121}$$

If the amplitude outcomes are $\pm A$ with equal probabilities,

$$\begin{aligned}\Phi_X(u_1,t_1;u_2,t_2)=\exp\Bigg\{&-\lambda\int_{t_1}^{t_2}\left[1-\cos\left(u_2\varphi(t_2-\alpha)\right)\right]d\alpha\\&-\lambda\int_0^{t_1}\left[1-\cos\left(u_1\varphi(t_1-\alpha)+u_2\varphi(t_2-\alpha)\right)\right]d\alpha\Bigg\}.\end{aligned} \tag{122}$$

The correlations arise from the second term. For stationary correlations the lower limit should be taken to $-\infty$. It's clear that the coefficients of $u_1u_2^3$ and $u_2u_1^3$ will be different, which indicates the absence of time reversal invariance. By Taylor expanding the characteristic function, we have access to all two-time correlators. Specifically,

$$\mathbb{E}\left(X(t_1)X^3(t_2)\right) = \lambda I_{1,3}^{-\infty,t_1} + 3\lambda^2 I_{1,1}^{-\infty,t_1}\left(I_{0,2}^{-\infty,t_1} + I_{0,2}^{t_1,t_2}\right), \tag{123}$$

$$\mathbb{E}\left(X^3(t_1)X(t_2)\right) = \lambda I_{3,1}^{-\infty,t_1} + 3\lambda^2 I_{1,1}^{-\infty,t_1} I_{2,0}^{-\infty,t_1}, \text{ where} \tag{124}$$

$$I_{n,m}^{t,t'} = \int_t^{t'} d\alpha\, \varphi^n(t_1-\alpha)\varphi^m(t_2-\alpha). \tag{125}$$

This result is general for any characteristic function of the form shown in Eq. (122), taking the lower limit to $-\infty$.