Quantum Machine Intelligence (2025) 7:17
https://doi.org/10.1007/s42484-025-00244-w

RESEARCH ARTICLE

Quantum variational rewinding for gravitational wave detection

Benjamin Rodrigues de Miranda[1,2] · Domenica Dibenedetto[1] · Niels Neumann[2] · Ward van der Schoot[2]

Received: 1 December 2023 / Accepted: 21 January 2025

Abstract
Gravitational waves carry information about far away phenomena which astronomers can use to form a better picture of the universe. Over the past few years, numerous gravitational waves have been detected using a method called matched filtering; however, this approach has two main drawbacks. First, it requires knowledge of the sought-after waveform beforehand. Second, it is infeasible for gravitational wave searches which are precise or cover a long time span. While attempts have been made to use classical machine learning to overcome these difficulties, the usage of quantum machine learning in this context is relatively unexplored. Quantum variational rewinding is a hybrid quantum-classical machine learning algorithm for time series anomaly detection. This work makes the first steps in investigating its potential for gravitational wave detection. Training and testing the algorithm was done without depending on synthetic data. In a noiseless simulated environment, quantum variational rewinding was able to pick out all confirmed gravitational wave signals from the background noise of a detector with perfect accuracy when filtering was applied before hand and was able to do so to a high degree of success without filtering. Anomaly detection with this algorithm was performed with two signals on quantum hardware as a proof of concept. Using Fisher information, both the capacity and the trainability of quantum variational rewinding are described so that future approaches can be compared in these regards. With new gravitational wave detectors under development, even more data will be generated for analysis in the future, compounding the existing bottlenecks in the search pipeline. As such, approaches which help tackle issues with the current approach to gravitational wave analysis are in demand, and the empirical results from this work assert that quantum computing should not be overlooked in this process.

Keywords Gravitational wave · Quantum computing · Quantum machine learning · Anomaly detection · Noisy intermediate-scale quantum

1 Introduction

Gravitational waves (GWs) are perturbations in spacetime caused by accelerating massive objects. Unlike electromagnetic radiation, they are not obscured by matter as they travel through the universe and thus offer unique insights into otherwise imperceptible phenomena. At the time of writing, there have been 90 confirmed GW detections (Abbott et al. 2021), with the first taking place in 2015.

Laser interferometers can be used to detect GWs by producing time-varying light intensity signals when GWs propagate through them. Unfortunately, the extreme precision of these devices means that noise of terrestrial origin also finds its way into the signals, which hinders the clarity of the GWs. A process called matched filtering (MF) is typically used to determine whether interferometer signals are caused by GWs. All GWs detected to date have been

Domenica Dibenedetto, Niels Neumann, and Ward van der Schoot contributed equally to this work

✉ Benjamin Rodrigues de Miranda
rodriguesdemiranda99@gmail.com

Domenica Dibenedetto
domenica.dibenedetto@maastrichtuniversity.nl

Niels Neumann
niels.neumann@tno.nl

Ward van der Schoot
ward.vanderschoot@tno.nl

[1] Advanced Computing Sciences, Maastricht University, Maastricht, Netherlands

[2] Applied Cryptography and Quantum Algorithms, Nederlandse Organisatie voor Toegepast Natuurwetenschappelijk Onderzoek, Hague, Netherlands

Published online: 08 February 2025

confirmed with MF, but it is not without drawbacks. These include the need for a template of the sought-after GW before it is detected, and a time complexity which can render the process infeasible for long or precise searches.

A recently proposed quantum-classical hybrid machine learning (ML) algorithm for time series anomaly detection, called quantum variational rewinding (QVR) (Baker et al. 2023), has potential in the GW detection domain. Thanks to its use of a constant depth variational circuit, it is suitable for currently available quantum computers, which are part of the noisy intermediate-scale (NISQ) quantum era. By framing GW interferometer readings as anomalous signals in otherwise non-anomalous background noise, the trained model can perform GW detection in linear time, an improvement over the quasilinear time complexity of MF. Also of significance is the ability of QVR to learn complex patterns with few parameters, while only requiring non-anomalous data to train on. The latter point is of interest because of the low number of confirmed GW signals, as approaches relying on positive examples depend on synthetic data for training, reducing their ability to flag GW signals of unknown forms.

The main contribution of this work is the use of an NISQ-era-compatible quantum algorithm for GW detection without dependence on templates. Evidence is provided of the strong performance of this approach on real data in both a simulated environment and on real quantum hardware. The following research questions are addressed:

- To what extent is quantum variational rewinding successful in classifying known gravitational waves as anomalies?
- How fast can quantum variational rewinding perform anomaly detection on laser interferometer time series data?
- Does quantum variational rewinding offer an advantage over classical approaches in gravitational wave detection, and if so, is this advantage inherently quantum?

Section 2 gives an overview of related approaches in which ML or quantum computing has been used to aid GW detection, followed by a brief background on GWs and their detection using MF in Section 3. Section 4 introduces concepts on which QVR is built on before the algorithm is formally introduced in Section 5. In Section 6, the data and their acquisition are described, after which Section 7 states QVR implementation details. Then, Section 8 describes experiments and reports their outcomes. Suggestions for why the quantum nature of QVR is important are made in Section 9 before results are discussed in Section 10. Sections 11, 12, and 13 go over limitations, provide concluding remarks, and put forward interesting avenues to explore in future work respectively.

2 Related work

Numerous classical ML approaches, and a quantum algorithmic approach, have been applied to GW detection. The common goal has been to improve detection rates and alleviate both the computational intensity and dependence on templates involved in searching for GW by means of MF.

In the work of Kim et al., artificial neural networks have been shown to improve the performance of MF for specific types of GW searches. This was done by considering properties of source objects as well as variables from the MF process to make classifications (Kim et al. 2015). The intention was for this to be used in combination with the current search procedure. While the approach put forward in this work should also be used in tandem with traditional approaches as discussed in Section 10, it can alleviate computational load by flagging interesting signals. Convolutional neural networks implemented by Gabbard et al. have performed well for classifying specific types of GW using synthetic signals (Gabbard et al. 2018), but the dependency on templates in the construction of GW signals implies prior knowledge of the sought-after waveforms. Work by Baker et al. demonstrated the potential of random forests in distinguishing GWs from background noise based on attributes of input signals (Baker et al. 2015). This approach worked well for certain types of GWs, but it was still necessary to model them before hand. Morawski et al. used a convolutional autoencoder to learn detector noise, thereby overcoming the need for templates and then detected GWs as anomalies by comparing reconstructed signal loss values to a threshold (Abbas et al. 2021). Results showed promise on synthetic GW data, but testing on real GW signals fell short; while the authors suggest that this is due to the test signals being from older observational runs, no results making use of more recent data are available.

Work by Dreissigacker et al. (2019) employed deep learning with a focus on continuous GW detection. This is a class of GW which is yet to be detected and is especially problematic for MF due to its low amplitude and long duration. Results on synthetic data are impressive, but again, the dependence on synthetic data is undesirable. An overview of ML for GW detection was given by Cuoco et al. (2020) who also discussed efforts in glitch characterisation and parameter estimation. A venture into the quantum realm was made by Gao et al. (2022) with the application of Grover's algorithm to MF for a theoretical speed-up, but the space complexity is still too great for current hardware.

3 Matched filtering for gravitational waves

Signal processing techniques are required to extract the relevant parts of the signals produced by GW detectors from

background noise (Jaranowski and Królak 2012). Matched filtering is the technique most widely used for this purpose. It looks for the similarity between a theorised signal called a template and an actual signal which is affected by background noise. Based on some underlying assumptions about the background noise, it allows a level of statistical certainty to be computed that a given incoming signal matches a template. Properties of MF which are relevant to this work are mentioned here, and a mathematical description is included in Appendix C. Understanding the details of how GWs are generated and subsequently picked up by hardware on earth are not required for following this work. For reference, an overview is given in Appendix A. Other approaches which have seen success are also briefly mentioned in this section.

Various types of GW detectors exist, with laser interferometers, which are the only type this work concerns, being the most popular. These instruments produce time-varying light intensity signals when GWs pass through (Davis et al. 2021), and because the perturbations caused by GWs are tiny even for highly catastrophic events, these instruments are required to be extremely precise. While this allows the influence of passing GWs to be picked up, it also means that the resulting signal is susceptible to noise of terrestrial origin. Such noise manifests itself as fluctuations in the resulting signal and can come from a wide array of sources simultaneously, ranging from environmental factors to properties of the hardware (Aasi et al. 2015; Davis et al. 2021). Fluctuations which are consistently present in the interferometer signal over time make up background noise, but when they are temporary they are known as glitches. Such atypical noise transients can resemble GWs (Cabero et al. 2019), and glitch signals with power comparable to those of GWs have appeared with a frequency of roughly one per minute in Laser Interferometer Gravitational-Wave Observatory (LIGO) detectors (Abbott et al. 2020). In Fig. 1, a background noise signal as well as two GW signals are displayed. The first GW wave is apparent just from visual inspection while the second is hidden in noise.

The most computationally expensive step in MF is the conversion of the input signal from the time domain into the frequency domain. This is done with the fast Fourier transform, which has time complexity $\mathcal{O}(N \log N)$ for a signal of length N (Cooley et al. 1967). In this case, the length of a signal is the sampling frequency multiplied by the signal's duration. The overall time complexity of MF is $\mathcal{O}(MN \log N)$, where M is the number of templates, due to the need to perform a search for each template.

Matched filtering has demonstrable success in GW detection. In fact, for an event to be published it has been a requirement that MF identifies the signal in at least two different detectors (Abbott et al. 2020). That being said, there are drawbacks. First, an accurate template is required. In circumstances where perfect knowledge of the form of the GW is attainable, this is a non-issue. Unfortunately, it is rare that there is enough knowledge about the GW source to construct an accurate template, and a search is done over a range of template parameters in reality. In addition, it means unexpected GW signals which do not match a template, or GW signals which do match a template but overlap with each other, will remain burried in the background noise. Second, the quasilinear time complexity of MF means that using it becomes infeasible for precise searches, searches for long signals, or a combination of both. This issue is compounded by the fact that the number of templates can be upward of 10^{12} (Abbott et al. 2019). It is worth noting that the amount of information available about the sources of GWs can vary,

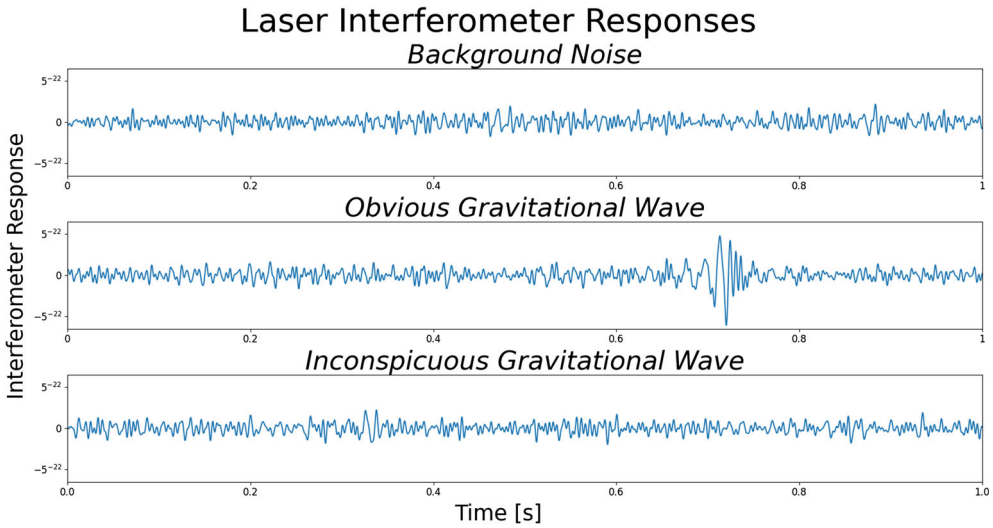

Fig. 1 Second long interferometer responses from LIGO Livingstone's third observing run. The top is from background noise, and the middle and bottom are from GW. These signals were subjected to the filtering described in Section 8

and so the severity of the computational load for data analysis is context dependent. The time complexity of MF is in and of itself not unmanageable, but when coupled with the sheer size of some current and future searches, it becomes problematic. Even for binary black hole coalescences, which are responsible for emitting all of the GWs detected to date (Abbott et al. 2021), factors like eccentric orbits can dramatically increase the difficulty of searches (The LIGO Scientific Collaboration et al. 2023).

A popular alternative approach makes use of multiple detector signals to avoid the need for templates and so facilitates the search for GWs of unknown form (Klimenko et al. 2016). For narrow searches such methods can be comparable in performance to MF; in fact, it was an initial low latency search with such an unmodelled approach that flagged the first detected GW within minutes of it reaching the detectors (Abbott et al. 2016). For more extensive searches, however, the time complexity of this approach (Necula et al. 2012) may become problematic just like in the MF case. It should also be noted that singular value decomposition has been investigated as a way to reduce the number of templates to search through (Cannon et al. 2010). With the ongoing development of more precise GW detectors, like the Einstein Telescope and the Laser Interferometer Space Antenna, the problems related to the computational cost of GW data analysis are expected to become more severe, increasing the demand for efficient algorithms (Couvares et al. 2021).

4 Variational simulation of hamiltonian evolution

Quantum variational rewinding is a hybrid quantum-classical ML algorithm which aims to approximate the evolution generated by a time-independent Hamiltonian with a variational circuit to detect anomalies in time series (Baker et al. 2023). The general idea is to learn the parameters of this circuit in a way that allows it to rewind the values of non-anomalous input time series back to some initial quantum state such that, upon measurement, there is a collapse into a predictable basis state. This way, the measurement of a different state after rewinding is an indication that the time series in question is anomalous. This technique has serious potential in the identification of GWs. Before delving into QVR in the next section, its foundational principals are explained here.

Quantum states evolve according to Eq. 1, which is Schrödinger's equation. The Hamiltonian $\hat{H}$ is an Hermitian operator which acts on the quantum state $|\psi\,(t)\rangle$ to bring about its time evolution at time t. $\hbar$ denotes Planck's constant. When $\hat{H}$ is time-independent, it leads to Eq. 2, which describes the continuous evolution of a quantum state from an initial time $t = 0$ to some time $t = T$ thereafter (Cresser 2011). Such evolution is necessarily unitary due the Hermitian nature of $\hat{H}$. Note that units are chosen such that $\hbar = 1$ for simplicity in Eq. 2, and this remains the case from here on out.

$$\hat{H}|\psi\,(t)\rangle = i\hbar\frac{d|\psi\,(t)\rangle}{dt} \tag{1}$$

$$|\psi\,(T)\rangle = e^{-i\hat{H}T}|\psi\,(0)\rangle \tag{2}$$

Evolution of a quantum system with a Hamiltonian as shown in Eq. 2 can be simulated with a variational circuit using the Suzuki-Trotter formula, which is shown in its first-order form in Eq. 3. The left-hand side corresponds to the operator on the right-hand side of Eq. 2. A summation term is present to take advantage of the fact that Hamiltonians can be decomposed into the sum of simpler Hamiltonians which are possible to simulate variationally. On the right-hand side, there is a product of matrix exponentiation terms, which are power series expansions. Each of these terms is evolution by one of the simpler Hamiltonians for time Δt. Taking the product of these terms after expanding them involves multiplying the H_j matrices with each other, and because matrix multiplication is generally not commutative, an error is incurred (Schuld and Petruccione 2021). The rightmost term captures this error and is dependent on Δt, which reflects that the source of this issue is the simulation of a continuous process with discrete time steps. Therefore, taking more time steps r with a smaller step size Δt improves the approximation. The number of time steps required to reach a time T is $r = \frac{T}{\Delta t}$.

$$e^{-i\sum_j \hat{H}_j T} = \left(\prod_j e^{-i\hat{H}_j\Delta t}\right)^r + \mathcal{O}\left(\Delta t^2\right) \tag{3}$$

Current quantum computers are part of the NISQ era (Schuld and Petruccione 2021), which is characterised by the presence of errors in quantum computations. There are a variety of causes, each contributing to less reliable results. Among these is quantum decoherence. This is a process by which information is lost due to the inability to maintain a quantum superposition for the amount of time required for a computation (Chuang and Nielsen 2010). The more quantum operations that are performed, the greater the presence of errors from these sources. Taking advantage of current quantum computers to implement variational algorithms therefore requires consideration of circuit depth. This is the maximum number of operators that need to be applied in succession in a quantum circuit. Evolving a quantum state using the Trotter-Suzuki formula has the drawback of depending on the repeated application of the gates implementing the desired Hamiltonian evolution for each time step. This, in combination with the fact that using smaller time steps gives a better approximation, means that quantum circuits can

become deep for sophisticated computations, which accumulates errors due to, among other causes, need to maintain superpositions for longer. Therefore, the error due to quantum decoherence can accumulate and become unmanageable for evolution beyond a given time.

Variational fast forwarding addresses this issue by performing a spectral decomposition of the evolution described by a Hamiltonian, as shown in Eq. 4. r applications of the Suzuki-Trotter equation can then be replaced by a single application of the spectral decomposition with the diagonal matrix raised to the power r, as shown in Eq. 5 (Cîrstoiu et al. 2020). Because the left-hand side is unitary, such a spectral decomposition always exists. The columns of W are the eigenvectors of the operator and D is a diagonal matrix containing the corresponding eigenvalues. By the spectral theorem, W, $W^\dagger$, and D are each unitary. With this approach, the depth of the variational circuit required to approximate the evolution brought about by the Hamiltonian remains constant as D^r captures the number of discrete time steps so that they do not need to be applied one by one. Thus, error caused by NISQ era devices is avoided, although the error incurred by discretising the evolution remains (Cîrstoiu et al. 2020).

$$e^{-i\hat{H}\Delta t} = WDW^\dagger \tag{4}$$

$$\left(e^{-i\hat{H}\Delta t}\right)^r = WD^rW^\dagger \tag{5}$$

5 Quantum variational rewinding

The ideas from variational fast-forwarding can be used in QVR. Again, a constant depth variational circuit is used to simulate the quantum evolution brought about by a Hamiltonian. Instead of considering forward evolution, however, QVR should be conceptualised as rewinding backwards through time. An overview of QVR is given here, with specific details given in Section 7. Discussion of how variational fast forwarding and QVR are linked to a concept called the no fast-forwarding theorem can be found in Appendix D, but this is only loosely related to this work.

The goal is to take each value of a time series $\vec{x}$ at time steps t_j in turn and rewind it back to some initial value at time $t = 0$ by applying Eq. 5 with $r = t_j$. In this setting, time goes backwards, so Δt is negative, causing t_j to be negative. The evolution of a quantum state with negative time is equivalent to the case where time is positive but the Hamiltonian is negated. The target Hamiltonian implemented by $WDW^\dagger$ in this rewinding setting is still valid because the negation of a Hermitian matrix is still Hermitian.

Given a set of time series, the goal is to determine which are anomalous by using this rewinding mechanism. That is, which of the time series do not follow a typical pattern. By rewinding values which belong to non-anomalous time series to the same initial quantum state, it is possible to tell which values belong to anomalous time series because they are rewound to a different initial state.

A quantum-classical hybrid ML scheme is used to this end, whereby a circuit is learned to approximate the backwards evolution brought about by a Hamiltonian capable of rewinding non-anomalous input values to a predictable quantum state. This Hamiltonian is unknown before training and will be referred to as the target Hamiltonian. By learning parameters of W and D, the target Hamiltonian is learned automatically. Training consists of a quantum component, wherein input time series values are encoded into a quantum superposition before being passed through $WDW^\dagger$ and measured using an observable operator, and a classical component, in which circuit parameters are updated based on derivative-free gradient calculations. Anomaly detection utilises the trained circuit, which now implements backwards evolution defined by the target Hamiltonian, by feeding in $(value, time)$ tuples of a time series one by one and determining whether their initial state corresponds to that expected of a non-anomalous time series. Quantum variational rewinding yields a quantum state for each input tuple. Determining whether a full time series is anomalous means averaging over all of the quantum states which come from its individual inputs.

To feed values into a quantum device, a quantum state needs to be created depending on this value. Encoding such an input value into a superposition can be seen as evolving the state from t_0 to t_j, a process which will be undone in the rewinding step. All qubits start in state $|0\rangle$, but as will be shown, this is not necessarily the initial state which will be targeted by the rewinding step. Equation 6 shows the encoding of the value of a time series x_i at time t_j with a unitary operation based on this value. The dependence of this encoding unitary on the value it is encoding is crucial, as it uses this value to create a quantum state representing the time series at time t_j. The rewinding step does not have access to this value and must rewind inputs from any non-anomalous time series at any time back to the same initial state using only t_j. The difference between these steps is that the latter is general enough to undo the forward evolution using only t_j when the input is non-anomalous. The rewinding step is the application of the Hamiltonian for time t_j via its spectral decomposition $WD^{t_j}W^\dagger$. Equation 7 shows the Hamiltonian approach, and Eq. 8 shows the variational approach.

$$|x_i\left(t_j\right)\rangle = U\left(x_i\left(t_j\right)\right)|0\rangle^{\otimes n} \tag{6}$$

$$|x_i\left(t_j\right), \vec{\alpha}, \vec{\epsilon}\rangle = e^{-i\hat{H}t_j}|x_i\left(t_j\right)\rangle \tag{7}$$

$$|x_i\left(t_j\right), \vec{\alpha}, \vec{\epsilon}\rangle = W\left(\vec{\alpha}\right)D\left(\vec{\epsilon}, t_j\right)W\left(\vec{\alpha}\right)^\dagger|x_i\left(t_j\right)\rangle \tag{8}$$

The W operations are parameterised by $\vec{\alpha}$ and the D operation is parameterised by $\vec{\epsilon}$, and it being raised to the power

of t_j means that t_j is also written as one of its parameters. $\vec{\alpha}$ and $\vec{\epsilon}$ are vectors containing trainable parameters which are updated during training, while t_j is the time corresponding to the input value. Because of this dependence on t_j, both the time series value and its corresponding time are fed into the circuit. Note that t_j determines how much backwards evolution is applied, in accordance with Eq. 8, but it does not fundamentally change the form of this evolution, and so it does not parameterise the target Hamiltonian like $\vec{\alpha}$ and $\vec{\epsilon}$ do. This can be seen by the absence of r in the parameters of $\hat{H}$ in Eq. 5, upon which Eq. 8 is based. Ideally, $W(\vec{\alpha})$ is a combination of single qubit general parameterised unitary gates G, the form of which is given by Eq. 9 up to a global phase for a set of parameters $\vec{\theta} = (\theta_1\ \theta_2\ \theta_3)$, and entangling gates (Schuld et al. 2020). Section 7 gives the exact form of these components.

$$G\left(\vec{\theta}\right) = \begin{pmatrix} e^{i\theta_2}\cos\theta_1 & e^{i\theta_3}\sin\theta_1 \\ -e^{-i\theta_3}\sin\theta_1 & e^{-i\theta_2}\cos\theta_1 \end{pmatrix} \tag{9}$$

Constructing D is a more involved process. While the structure of W constrains it to be unitary for any parameterisation, D must be diagonal, and so it is not unitary for an arbitrary selection of diagonal entries. For this reason, the construction in Eq. 10 is used where M is a diagonal matrix with entries $\vec{\epsilon}$. This forces D to be unitary regardless of its parameters.

$$D\left(\vec{\epsilon}, t_j\right) = e^{-iM(\vec{\epsilon})t_j} \tag{10}$$

Consider now two sine waves which are identical except for the fact that they are out of phase with each other. Letting either of these be a non-anomalous time series implies that the other one is as well because they change in the exact same way over time. It should be possible, then, to rewind all of their values to the same initial state so as to classify them both as non-anomalous. Choosing some t_j, the QVR algorithm will rewind a value from each using the same time interval according to this value, which may result in different initial states. Overcoming this would require choosing a different t_j for one of the time series to account for the phase difference. After evaluating the matrix exponential in Eq. 10 to compute D, t_j ends up multiplying each element of $\vec{\epsilon}$. What this means is that it is possible to see the desired effect of modifying t_j by instead modifying $\vec{\epsilon}$. This is achieved by sampling each of the diagonal elements $\vec{\epsilon}$ of M from their own normal distributions. This entails having a distribution for each diagonal entry in M. To sample multiple M matrices, the elements at each diagonal entry are all sampled from the normal distribution which corresponds to that entry. These distributions are defined as $\vec{\epsilon} \sim \mathcal{N}(\vec{\mu}, \vec{\sigma})$, where the vector of means μ and the vector of standard deviations σ both have a length equivalent to the number of diagonal entries in M.

Strictly speaking, the elements of $\vec{\epsilon}$ are not trainable parameters; rather the $\vec{\mu}$ and $\vec{\sigma}$ defining the distributions from which these are sampled are. That means that for a set of parameters, the circuit structure is not fixed because it depends on sampling from these trainable distributions. Therefore, multiple different circuits can be generated from the same set of parameters. Combating the difficulties related to signals displaying similar behaviour at different times is done by generating several circuits for each input tuple, each differing only in its composition of D due to the resampling of $\vec{\epsilon}$. A single input $(value, time)$ tuple is fed through each one of these circuits and the outcome is averaged. When values from a time series are not rewound to the initial state expected of inputs from non-anomalous time series, it should be because behaviour is present in the signal which does not fit the trends of non-anomalous signals. Because of the resampling of D, there is a lower chance that a signal which does not exhibit such behaviour is flagged as an anomaly because non-anomalous signals which are out of phase with each other are accounted for.

Of course, the D matrices must be similar enough to each other so that the circuits still implement rewinding procedures which are similar to each other. How different the D matrices are from each other will depend on what the optimisation routine determines is an appropriate balance between having enough variety to be able to account for patterns appearing at different times and having enough similarity to facilitate a consistent rewinding process. Regularisation is applied on the standard deviations of the distributions to control this balance.

After encoding and subsequent rewinding, the quantum state is measured in the bases defined by a trainable observable. A consequence of this is that the superposition can collapse to a basis state not in the original $\{|0\rangle, |1\rangle\}$ basis. In this way, rewinding is done to ensure that non-anomalous time series inputs cause a collapse to a predictable state defined by this observable. During training, a loss value is computed for each $(value, time)$ input tuple based on the expected value of the observable acting on the superposition. Optimising the circuit involves computing the gradients of this expected value with respect to each parameter. On quantum hardware, standard approaches to gradient computations in classical ML, like backpropagation, are not applicable (Schuld and Petruccione 2021). To overcome this, gradients can be computed by parameter shift rules (Schuld and Petruccione 2021) or derivative-free optimisation routines can be used (Baker et al. 2023). After training, a time series is assigned an anomaly score based on the average loss across all of its $(value, time)$ input tuples. If the anomaly score of a time series equals or exceeds a tunable threshold it is flagged as an anomaly. The whole process has a time complexity of $\mathcal{O}(N)$ when run on a quantum computer.

Working directly with the parameters of the spectral decomposition of the unitary implementing the evolution brought about by the Hamiltonian avoids having to implement a non-elementary unitary after each parameter update. Of course, it is possible to learn the parameters of the unitary operation implementing the rewinding directly, but were this the case then every time the parameters were updated, a new circuit would need to be constructed to implement it as a series of elementary gates, which would invoke unnecessary overhead. The W and $W\dagger$ operations in the formulation in Eq. 4 already consist of only elementary operations by design and the diagonal unitary D can be approximated by elementary gates efficiently (Welch et al. 2014). The resulting circuit has constant depth and avoids the overhead related to the implementation of parameterised non-elementary unitaries.

6 Data

All data were collected as 4096 Hz one-second signals from the LIGO Livingston interferometer during its third observing run which spanned from 01/04/2019 to 27/03/2020 using the **gwqml** package (Gobeil 2023; Nitz et al. 2023; Macleod et al. 2021; Christ et al. 2018). All signals, therefore, were comprised of 4096 time stamps. Non-anomalous data were signals caused by the background noise of the detectors in the absence of GWs. These were sampled starting at random times between 100 and 1000s after confirmed GW wave signals, which ensured that no confirmed events were sampled as non-anomalous data. Anomalous data were signals labeled by the LIGO-Virgo-KAGRA collaboration as auxiliary, marginal, or confident, categories which reflect increasing levels of confidence that an incoming signal was caused by a GW. These were positioned off-centre uniformly at random between -0.25 and 0.25 seconds to imitate the uncertainty of the position of potential GW signals in the treatment of real interferometer readings. The signals ranged from as short as 0.2 seconds (Abbott et al. 2016) to over 100 seconds long (Abbott et al. 2017). Therefore, some anomalous signals contained full events while others only contained parts of events.

A total of 5000 background noise signals made up the non-anomalous data. 4000 of these were used for training, 78 made up a validation set which was used for threshold tuning, and an additional 78 were used for testing. The validation and testing data sets were of size 78 because this is the total number of GW signals which made up the anomalous data. Testing was done by attempting to separate all 78 of the anomalous signals from the 78 non-anomalous signals in the testing set. Between training and testing, validation data was used to set the threshold. One way to do this was to compute anomaly scores for a number of anomalous and non-anomalous signals and then perform a grid search across various thresholds, choosing the one which was best able to separate the anomalies from the non-anomalies. Another option was to set the threshold using statistical properties of only the anomaly scores of the non-anomalous validation data. For example, the threshold could have been set using the scaled standard deviation of these anomaly scores, but seeing as this choice lacked theoretical motivation, the decision was made to tune the threshold with the first option, a grid search. This was done by computing the anomaly scores for anomalous and non-anomalous signals in the validation set and then computing the accuracy in separating them using 1000 different thresholds ranging from zero to the maximum computed anomaly score. The threshold which resulted in the best accuracy was chosen. Because the testing set was already of limited size, the same anomalous signals were used for threshold tuning and for testing instead of reducing its size further by splitting the data into two sets as was done with the non-anomalous signals. While this decision had no impact on the training process or the computation of anomaly scores, there are consequences related to actually flagging anomalies. Section 11 discusses these. Blind injections, which are fake GW signals produced to test data analysis capabilities, were not included at any stage, except for the injection of a signal intended to emulate a continuous GW, which was present for most of the third observing run at a low enough amplitude to not interfere with the analysis of other data.

7 Implementation

Armed with a description of QVR and the data to use it on, the specific implementation details can now be described. Recall that the goal was to learn the gates of a quantum circuit to rewind non-anomalous signals to the same initial states optimally. To do so, gate parameters had to be initialised, and then, iteratively, an output measured, a loss value computed, and the parameters updated. These are the steps described here. For an illustration of how QVR is used to detect GWs, refer to Fig. 2.

An advantage of QVR is that various components can be adapted to suit the problem at hand. A summary of the algorithm design can be found in Table 1, which shows that much of the originally proposed structure was left unaltered. Table 2 describes the trainable parameters and their initialisation.

For GW detection, two qubits were used; one to encode the input values and another as an additional output register. Angle encoding was used on the first qubit, which corresponds to the top wire in both Figs. 3 and 4, to create a quantum representation of the input data, while the second qubit was left unaltered. Angle encoding was chosen

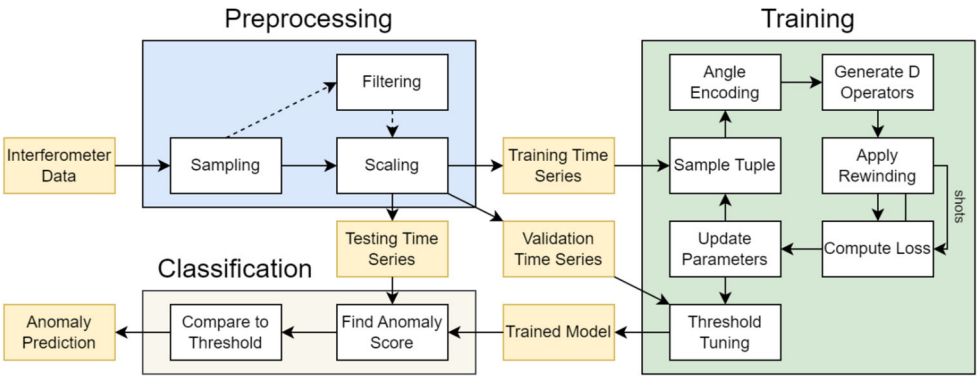

Fig. 2 How QVR is used to detect GWs

because it facilitates expressive models (Schuld and Petruccione 2021). W was implemented in three layers of the form shown in Fig. 3 and $W^\dagger$ was its conjugate transpose. Each layer consisted of the single qubit rotation gates X, Y, and Z in succession to implement the rotation given by Eq. 9, followed by a single controlled not gate, with the angle encoded qubit as the control, to generate entanglement.

D was approximated using a process involving Walsh series (Welch et al. 2014), resulting in the circuit shown in Fig. 4. In this figure, γ_1, γ_2, and γ_3 are the diagonal entries of D, which are not the same as $\vec{\epsilon}$, which define M upon which D is based. The final depth in terms of single qubit rotation gates controlled not gates, and a measurement gate was 30. Contributions to this depth from the different components were as follows. One from angle encoding, four from each layer of W and $W^\dagger$, four from D, and one from the measurement observable.

Table 1 Summary of QVR components

Component	Implementation
Number of time series per batch	10
Number of times per batch	10
Number of distributions	3
Regularisation hyperparameter	15
Number of qubits	2
Encoding	X rotation on the first qubit
Eigenvector operator	Three layers of Fig. 3
Diagonal operator	Walsh operator approximation in Fig. 4
Optimisation	Powell's conjugate direction method
Loss function	Eq. 14
Anomaly score	Average contribution from Eq. 13
Threshold setting	Grid search
Observable	Eq. 11
Number of shots	256

Both of the $n = 2$ qubits were measured using the observable $\hat{O}$ shown in Eq. 11, which is the same as taking the total energy resulting from a Z basis measurement on each qubit, denoted by σ_z, scaled by a factor of $\frac{1}{2}$. Then the distance between this output and a trainable parameter η was taken over a number of shots. What was left was a distance measure Ω representing how far the expected value of the measurement was from η, something which is shown in Eq. 12.

$$\hat{O} = \frac{I^{\otimes 2} \otimes \sigma_z + \sigma_z \otimes I^{\otimes 2}}{2} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \tag{11}$$

$$\Omega\left(x_i\left(t_j\right)\right) = \eta - \left\langle x_i\left(t_j\right), \vec{\alpha}, \vec{\epsilon} \left| \hat{O} \right| x_i\left(t_j\right), \vec{\alpha}, \vec{\epsilon} \right\rangle \tag{12}$$

Each individual $(value, time)$ input contributed to the loss value used to perform the parameter updates. The square of the average distance in Eq. 12 across several circuits, differing due to the resampling of $\vec{\epsilon}$, was this single point contribution and is shown in Eq. 13. The training was done in mini-batches B of size $N = 100$, which were constructed

Table 2 Summary of QVR trainable parameters

Parameter	shape	Description	Initialisation
$\vec{\alpha}$	(3, 2, 3)	Angles for X, Y, and Z rotations for each qubit in each layer of W	Uniformly at random in the range $[0, 2\pi)$
$\vec{\mu}$	(1, 3)	Means of the normal distributions from which $\vec{\epsilon}$ is drawn	Uniformly at random in the range $[0, 2\pi)$
$\vec{\sigma}$	(1, 3)	Standard deviations of the normal distributions from which $\vec{\epsilon}$ is drawn	Uniformly at random in the range $[0, 2\pi)$
η	(1)	Value from which distance to $\hat{Z}$ is measured to calculate loss	Uniformly at random in the range $[-1, 1]$

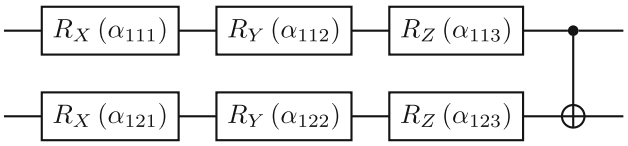

Fig. 3 A single layer of the eigenvector operator W

by sampling ten-time series from the training data uniformly at random, and, for each one of these, sampling ten stamps uniformly at random. It is for this reason that the loss function, shown in Eq. 14, takes multiple time series. The loss across a whole mini-batch was the average of the losses of the individual inputs with a regularisation term given by Eq. 15 added to penalise high standard deviations in the sampling distributions for $\vec{\epsilon}$. This regularisation expression was not modified from the original formulation (Baker et al. 2023).

$$\mathcal{C}_{single}\left(x_i\left(t_j\right)\right) = \underset{\vec{\epsilon}\sim\mathcal{N}(\mu,\sigma)}{\mathbb{E}} \Omega\left(x_i\left(t_j\right)\right)^2 \tag{13}$$

$$\mathcal{C}_{batch}\left(\overrightarrow{x\left(t\right)}\right) = \frac{1}{N}\sum_{x_i\left(t_j\right)\in B_N} \mathcal{C}_{single} + R \tag{14}$$

$$R = \frac{1}{\pi Q}\sum_{m=1}^{Q} \arctan\left(2\pi\tau\left|\sigma_m\right|\right) \tag{15}$$

Ten diagonal M matrices were sampled from $Q = 3$ normal distributions per input. Therefore, each $\vec{\epsilon}$ had three elements which made up the first three diagonal entries of one of the M matrices, with the last entry being zero. Equation 15 shows that the regularisation term took an average across the arc tangents of the standard deviations σ_m of the normal distributions from which $\vec{\epsilon}$ was sampled. The hyperparameter τ was set to 15. The number of diagonal matrices and the values for Q and τ were as originally proposed (Baker et al. 2023). Regularisation ensured that the D operations were similar enough to each other that the circuit as a whole still implemented roughly the same rewinding operation. After training, each input time series was assigned an anomaly score which was the average of the individual contributions $\mathcal{C}_{single}$ of all of its $(value, time)$ tuples.

Optimisation was done with Powell's conjugate direction method, which is not dependent on derivatives (Powell 1964). It works by evaluating the loss function along vectors in parameter space to find minima. Here, the search was bounded for each parameter by the same range that

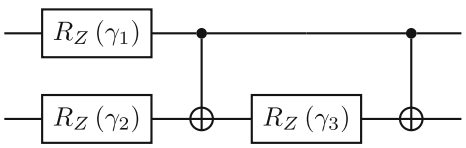

Fig. 4 The diagonal operator W

parameter was initialised with, which can be seen in Table 2. While all training was done classically here, the method being derivative-free means training could also be done on a quantum computer without needing to adapt the optimiser. This is relevant because recent evidence suggests that learning the noise of a quantum computer during the training of a quantum ML algorithm may be advantageous (Liao et al. 2023).

Time series values were normalised to be between zero and 2π before training. As opposed to doing this across all data, normalisation was performed on the different series at each time stamp individually. That is, for each of the 4096 measurement times, the values of all series at that time were grouped and normalised. As a result, at each time stamp the time series values range between zero and 2π, so what is learned is not the extremity of the signal values, but how spread out they are within this range. The non-anomalous data that was learned by QVR was a combination of stationary Gaussian noise, small fluctuation patterns, and glitches. In pure stationary Gaussian noise, the spread of values at any given time is even across multiple signals. Additional fluctuations in the detector readings then condense the spread around specific values. Where the dense regions are at consecutive time stamps is what is learned. The actual times were also normalised such that the first measurement corresponds to time $t = 0$ and the final measurement corresponds to time $t = 2\pi$.

8 Results

Three sets of experiments were performed. The first and second tested QVR on filtered and unfiltered GW data respectively, both in a simulated environment. Signals which were filtered were passed through both a band-pass filter with cut-offs at 50 Hz and 250 Hz and a notch filter with notches at −60 Hz, 120 Hz, 180 Hz, and 240 Hz, which removed some infeasible frequencies. Figure 5 shows that the patterns become obscured by fluctuations which are orders of magnitude bigger when no filtering is applied, as evidenced by the large difference in scale between the two responses to the same GW. In the final experiments, QVR on filtered GW data was implemented on a quantum computer. The results which follow are all rounded to three significant figures. When multiple runs were performed, results were selected to be displayed based on how informative they were. These graphs therefore show results from different runs. In general, however, the patterns were similar across the runs in an experiment. All code, with the QVR implementation being heavily based on the original implementation by Baker et al. (2023), has been made publicly available (Github Repository 2023).

Pennylane's (Bergholm et al. 2022) **lightning.qubit** was used to perform a noiseless simulation in which QVR was trained on filtered non-anomalies and validated and tested on

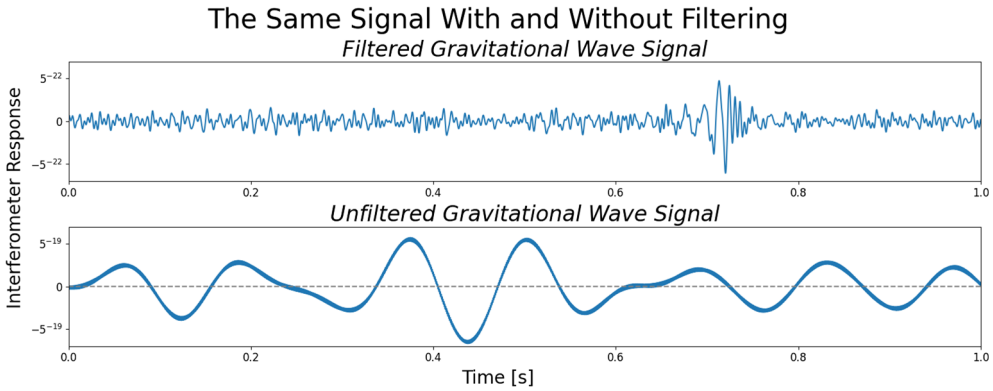

Fig. 5 The interferometer response to the same GW with and without filtering. The filtered signal is also shown as a dashed line in the bottom graph

both filtered non-anomalies and filtered anomalies. A modified version of five-fold cross validation was used. Namely, the validation and testing sets consisted of the same anomalous data while for the non-anomalous data they differed. This decision was made to avoid taking some of the anomalies from the testing set, which was already limited due to the low number of available GW signals, to put them in a validation set. Section 6 describes the data partitioning which was performed for each run. Aside from training on different data partitions, cross validation also meant that the parameters were reinitialised, preventing the misrepresentation of results which were caused by a specific initialisation rather than by general performance. It is from these partitions that 2000 mini-batches were sampled for training in each run.

Training loss across the mini-batches is shown in Fig. 6 for two runs. These were selected to display the difference in convergence pattern between a model that did not start with near optimal parameters and one which did, a difference arising from random parameter initialisastion. The models ended up at around the same training loss after 2000 mini-batches, and in fact all of the models converged to training losses within 0.009 of each other. This is a measure of how close the rewound states were to the initial states after training. Because units were chosen such that Planck's constant was 1 in Eq. 1, this number on its own does not hold any significance. Looking at the range of loss values expressed during the training after unlucky initialisation in Fig. 6, it becomes apparent that each model was able to converge to similar loss values regardless of the initial parameters.

Categories have been created to flag when there are excessive amounts of noise present in LIGO data (Abbott et al. 2023), which leads to an especially high concentration of glitches (Aasi et al. 2015). Specifically, these target periods of time when detector data should not be used for GW searches of certain types. One such category indicates segments of data for which it is infeasible to search for GWs produced by events called compact binary coalescences. These make up all confirmed instances of GW signals found to date (Abbott et al. 2021), and thus define all of the anomalies in the test data. Data in this category is marked as unsuitable for analysis, but was not excluded from the sampling of non-anomalous training, threshold tuning, and testing data in this work. Therefore, a small amount of the used data was unsuitable for searching for the type of GW signal which QVR was trying to flag as anomalous in these experiments. Of the 5000 non-anomalous signals, 57 overlapped with an

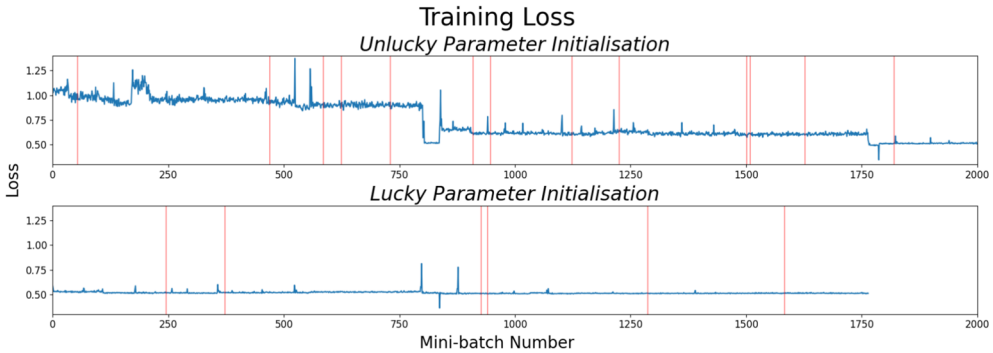

Fig. 6 Loss per mini-batch during training for two of the five runs. Bars indicate the presence of at least two unsuitable signals in a mini-batch. Training of the run corresponding to the bottom graph was terminated early because the loss was not decreasing

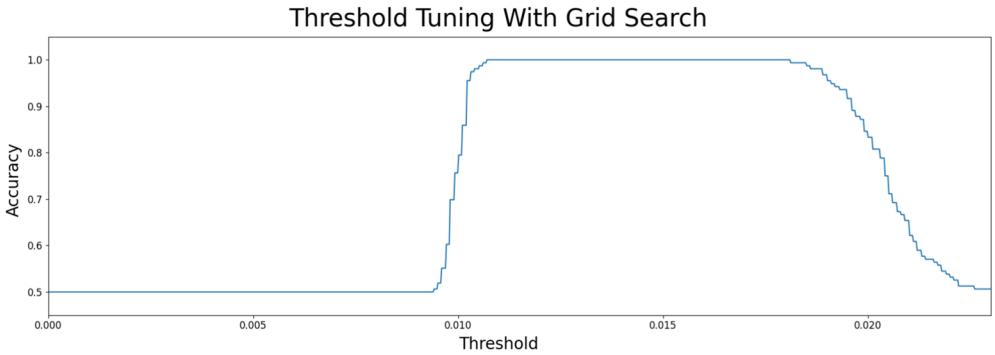

Fig. 7 Accuracy on validation data at different thresholds

unsuitable segment in the LIGO Livingstone detector. Quantifying the presence of unsuitable data during training was done by counting how many of the one second long signals out of the ten in each mini-batch fell within segments of detector data which were in this category. Mini-batches which contained at least two signals sampled from an unsuitable segment are marked with a red bar in Fig. 6. That is, the red bars mark when at least 20% of the data in a mini-batch was unsuitable. In these two runs, no mini-batch contained a higher than 30% concentration of unsuitable signals.

Figure 7 illustrates the performance of the model for different thresholds. For thresholds set too low, all validation data was classified as non-anomalous and for thresholds set too high, all validation data was classified anomalous. This is because the validation data was equal part non-anomalous and anomalous, so half of the examples were classified correctly in each case, explaining the flattening at an accuracy of 0.5. Accuracy was an appropriate metric to gauge model performance because there was no class imbalance. This held for all threshold tuning and testing in all of the experiments in this work. There were a range of thresholds for which maximum accuracy was achieved on the validation data, and the threshold was set at the midpoint of these. For the unlucky model, this was at 0.0144. There was no special treatment of unsuitable data during threshold tuning.

Both runs depicted in Fig. 6 achieved 100% accuracy on their respective test data, which is depicted for the s model by the full separation in Fig. 8. In four of the five runs, this was the case, but the other only achieved an underwhelming accuracy of 0.596. This is also shown in Fig. 8. Note that the anomaly scores are represented on different scales for clarity. The boxes cover the first to third quartiles of the anomaly scores, with the horizontal lines they contain showing the median. The whiskers indicate 1.5 interquartile ranges on both sides of the medians, and the circles beyond these are data which fell outside of this range. Finally, the dashed blue lines represent the thresholds of the respective models. Clearly, the successful model was better able to distinguish between anomalies and non-anomalies in calculating their anomaly scores, as indicated by there being no overlap compared to the large overlap for the unsuccessful model. In addition, all non-anomalies were contained within the whiskers, unlike the unsuccessful model. The maximum number of signals which at least partially overlapped with unsuitable detector segments in the any of the testing sets was two. Therefore, no testing set contained a significant number of non-anomalies which were unsuitable for analysis.

Next, experiments were done under the same conditions except that none of the signals underwent filtering before hand. An example of each of the steps of QVR applied to

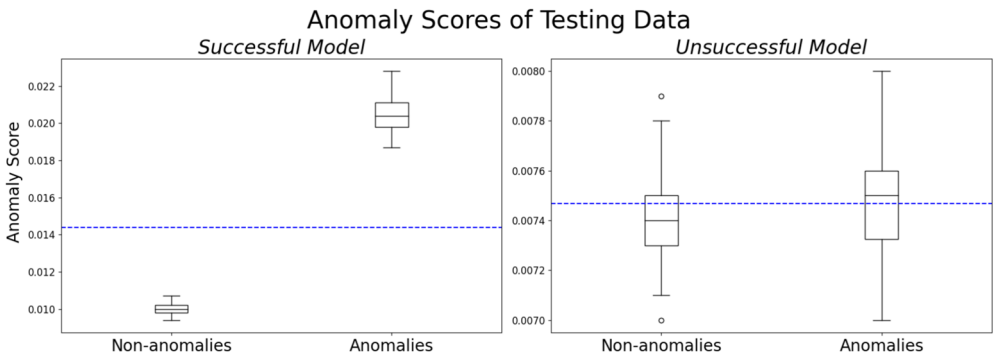

Fig. 8 Anomaly scores from applying a trained model to testing data

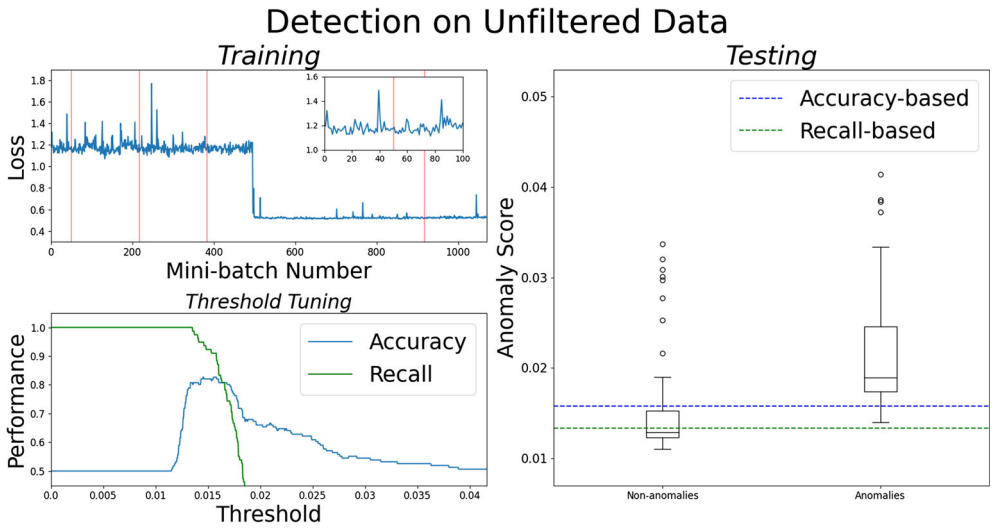

Fig. 9 All phases of QVR on unfiltered GW data. Training was terminated early because the loss was not decreasing in the top left, which also shows an enhanced version of the training loss for the first 100 mini-batches

unfiltered GW data are shown in Fig. 9. The best accuracy achieved across the five runs was 0.833. Recall is a more appropriate metric in cases where it is important to detect all of the GWs, even if it is at the expense of some false positives. Perfect recall can be achieved by classifying all examples as anomalies, so it was not used on its own for threshold tuning or model testing. Instead, accuracy was used under the condition of perfect recall. This entailed looking at the set of thresholds for which perfect recall was achieved, meaning all GWs were classified as anomalies, and then maximising accuracy across this set. Figure 9 shows performance during threshold tuning in terms of both accuracy and recall and shows the optimised threshold in terms of both accuracy and accuracy under perfect recall during testing. Table 3 details the performance of the best models, both filtered and unfiltered, as measured by highest regular accuracy on the testing data after training and then by the lowest achieved loss to break ties. Each run on unfiltered data resulted in classifying some non-anomalies as anomalies, but at most one of these per run was from a period of unsuitable data.

Unsurprisingly, the best models had lower than average final loss values, and the models using filtered data, which generally outperformed the models using unfiltered data, had a lower average loss value. In Table 4, the initialised and trained values for η are given. All values ended up within the range $(-0.5, 0.5)$ despite the initialisation range being $[-1, 1]$. Furthermore, in eight out of the 10 total runs, the final value was closer to zero than the initial value.

The learned parameters of the model which performed best on filtered data were hard-coded into a **Qiskit** (Qiskit contributors 2023) circuit, which was then submitted to the quantum computer ibm_kyoto to compute the anomaly scores of one filtered non-anomalous and one filtered anomalous time series. This conversion was made to allow for the batching of jobs from within a session to reduce queuing times. Twirled readout error extinction was applied by using **Qiskit**'s default resilience level of one. Using ibm_koyto, a 127 qubit computer, for computations on two qubits was inefficient due to the large number of idle qubits but it was used

Table 3 Summaries of the best and performing and average performance of the QVR runs on filtered and then unfiltered data

	Filtered		Unfiltered	
	Best model	Model mean	Best model	Model mean
Final loss	0.509	0.510	0.514	0.523
Threshold	0.0106	0.0131	0.0131	0.0196
Accuracy	1	0.919	0.904	0.75
Alternative threshold	0.00894	0.0105	0.0129	0.0155
Alternative accuracy	1	0.9	0.904	0.671

Alternative columns indicate what the results are under the condition that recall is perfect

Table 4 Each run's value for η before and after training for experiments on both the filtered and unfiltered data

	Filtered		Unfiltered	
	Initial η	Final η	Initial η	Final η
First run	0.4926	0.4704	0.4719	0.4344
Second run	−0.6432	−0.0311	0.4375	0.3211
Third run	−0.1072	0.0537	0.0952	0.1738
Forth run	0.2265	0.2319	−0.8090	0.0257
Fifth run	−0.8983	−0.0118	−0.9861	−0.1903

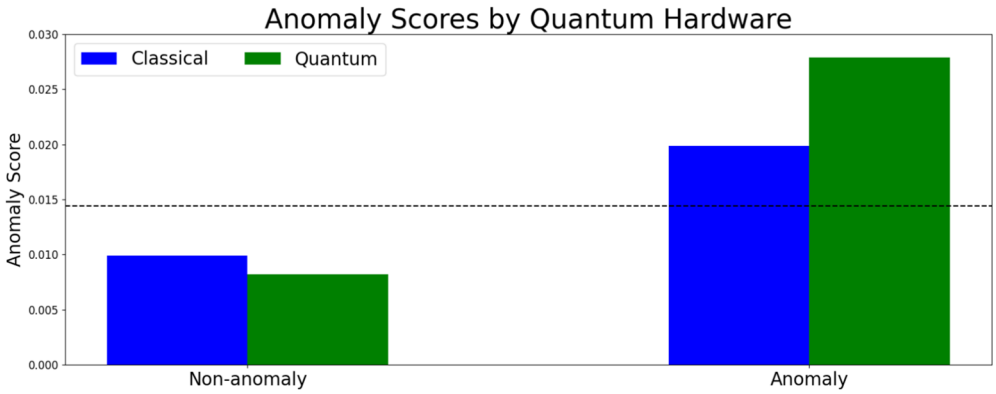

Fig. 10 Anomaly scores of one anomalous and one non-anomalous signal, both filtered, as computed in a noiseless simulation and by the ibm_kyoto quantum computer. The dashed line shows the threshold calculated in a simulation

because of limited access to quantum hardware. For this same reason, only 70 $(value, time)$ tuples were averaged over to compute the anomaly scores in the quantum case. These were selected from the input signal at evenly spaced intervals. The classical computations made use of all 4096 tuples.

No further error mitigation techniques were applied. Individual contributions of the $(value, time)$ inputs were computed on quantum hardware before averaging was done locally. The times taken by the quantum processing unit, ignoring classical data transfer, were 496 seconds and 503 seconds for the non-anomaly and the anomaly respectively. These times are how long the quantum processing unit was dedicated to computations. Figure 10 shows that the threshold computed in the simulation was sufficient to separate the non-anomaly from the anomaly in the quantum setting.

9 The significance of a quantum approach

It is important to address whether the quantum nature of QVR is essential for attaining the results presented in Section 8. If time series anomaly detection for GW detection using classical computation is sufficient, it becomes difficult to justify the use of quantum hardware for this purpose, especially considering the high cost and scarce availability of quantum computers and the fact that the time complexity of this QVR is the same as that of classical neural networks. Showing that the algorithm is necessarily quantum is difficult because the paradigms of quantum and classical computing are fundamentally different. As such, no fair comparison can be made between the same algorithm implemented on a classical computer and on a quantum computer. Because classical computers can simulate quantum computers, any quantum algorithm can also be run classically, but with a significantly higher time complexity. Classically, a state vector can be evolved by a matrix similar to how a quantum state is evolved by a Hamiltonian. An important distinction is that the former depends on matrix algebra, but for the latter this is just a way of describing the evolution which itself is much faster than performing the calculations associated with this description.

In comparing the QVR to the classical ML approaches mentioned in Section 2, it appears to perform better based on the evidence from Section 8. Seeing as these comparisons are based on different data, they should be taken with a grain of salt. Explaining the cause for apparent performance advantage can be done in part by the concepts of capacity and trainability. Respectively, these describe the extent of a ML model's abilities to express a variety of functions and to actually learn those functions by training. It has been shown that certain types of quantum neural networks are stronger in these regards than their classical counterparts, as measured by the spectrum of the Fisher information matrices of the models (Abbas et al. 2021). Based on this, it is plausible that the quantum nature of QVR translates to better trainability and capacity.

This idea can be used to explain discrepancies in the performance of quantum and classical ML approaches to GW detection. Unfortunately, doing this requires comparing the Fisher information matrix spectra of the models in question. The implications for this work are that a basis for comparison can be laid out, and indeed has been, but determining whether QVRs apparent success compared to classical approaches can be attributed to these principals relies on future work. Readers are encouraged, therefore, to make such comparisons to the results in this section.

The Fisher information matrix treats a model as the conditional probability distribution in Eq. 16. That is, for a parameterisation $\vec{\theta}$, the probability of observing the label y associated with an input x is conditional on x and depends on the probability of selecting the pair (x, y) form a prior distribution p. In this case, the parameter vector θ contains all the trainable parameters in Table 2. With this, the Fisher

information matrix is formulated as in Eq. 17 and is approximated by the empirical Fisher information matrix using Eq. 18.

$$p\left(x, y; \vec{\theta}\right) = p\left(y|x; \vec{\theta}\right) p\left(x\right) \tag{16}$$

$$F\left(\vec{\theta}\right) = \mathop{\mathbb{E}}_{(x,y)\sim p}\left[\frac{\partial}{\partial\vec{\theta}} \ln p\left(x, y; \vec{\theta}\right) \frac{\partial}{\partial\vec{\theta}} \ln p\left(x, y; \vec{\theta}\right)^T\right] \tag{17}$$

$$F\left(\vec{\theta}\right) = \frac{1}{k}\sum_{j=1}^{k} \frac{\partial}{\partial\vec{\theta}} \ln p\left(x_j, y_j; \vec{\theta}\right) \frac{\partial}{\partial\vec{\theta}} \ln p\left(x_j, y_j; \vec{\theta}\right)^T \tag{18}$$

In Eqs. 17 and 18, a partial derivative is taken for all parameters to produce a column vector. Multiplying this column vector by its transpose leaves a matrix. Pairs $\left(x_j, y_j\right)$ were drawn uniformly at random from the training data $p\left(x, y; \vec{\theta}\right)$, with $y \in \{0, 1\}$, where zero or one indicated that the x associated with y was from a non-anomalous or anomalous signal respectively. The Fisher information matrix depends on how the output changes with movements in parameter space. The phase of QVR which concerns movements in parameter space is during training, which means that computing its Fisher information matrix involves only data used in training, which was non-anomalous. As such, of interest is the probability that the model associates x with the output $y = 0$ for each sampled x. A few adjustment were necessary to convert QVR into a form facilitating a Fisher information matrix.

First, it had to be framed as a statistical model. This was done by feeding the squared distance measure from Eq. 13 through Eq. 19, which is depicted in Fig. 11. Because the squared distance measure was guaranteed to be greater than one, the new output was a probability value between zero and one. With this addition, the contribution of an individual input tuple to the anomaly score could be assigned a value according to the model's confidence that the label belonging to that tuple was zero. Classifications were made by the original QVR model by comparing the average individual contributions to a threshold. During training, therefore, the individual contributions generally had to be less than the threshold because the data was non-anomalous. The post-processing function, Eq. 19, was centered at such a threshold ζ so that the further below the threshold an individual contribution was, the higher the output probability of it being classified as a contribution from a non-anomalous signal, represented by the label zero. Conversely, high individual contributions indicated a low probability that the outcome was zero. A contribution on the threshold meant an output of 0.5 to reelect maximum uncertainty in the label of the training example it came from. Knowledge of the threshold was required before hand to do this, and one of the thresholds, $\zeta \approx 0.0144$, for a model which achieved perfect accuracy on unfiltered data was selected. Recall that these new outputs are to reflect the probability that the model associated the correct label with an input training example. Because all training examples were non-anomalies, the output of this step reflects the model's confidence that it has correctly classified a non-anomalous signal as such assuming knowledge of the threshold before hand. This process is similar to how the softmax function is applied in classical ML to convert a model's output into a probability value.

$$f\left(x\right) = \frac{1}{\frac{x}{\zeta} + 1} \tag{19}$$

To compute the empirical Fisher information matrix with Eq. 18, the partial derivative of the natural log of Eq. 19 with respect to each parameter $\theta \in \vec{\theta}$ needs to be taken. Because the results reported in this work made use of gradient-free optimisation, there was no access to these partial derivatives. A second modification was therefore made to use a parameter shift to compute the distance measure instead of Powell's method. The partial derivative with respect to a parameter θ using a parameter shift is shown in Eq. 20, where $[c_0, a_0, s_0] = \left[\frac{1}{2}, 1, \frac{\pi}{2}\right]$ and $[c_1, a_1, s_1] = \left[-\frac{1}{2}, 1, -\frac{\pi}{2}\right]$.

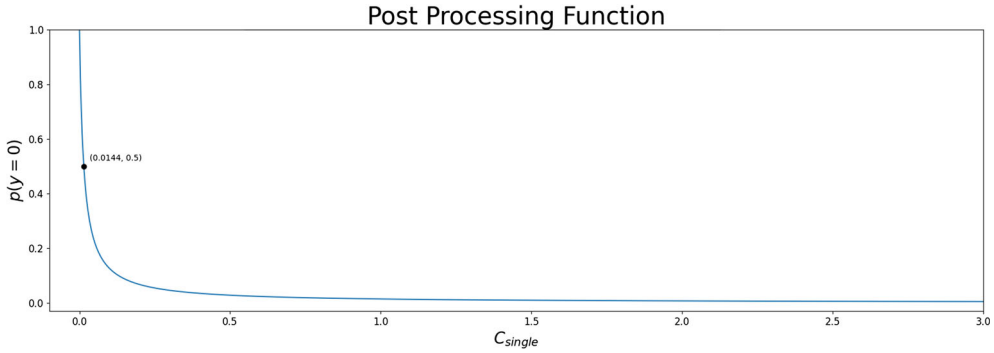

Fig. 11 Post processing function to convert model output into a probability value

These are the default values for parameter shift in **Pennylane**.

$$\frac{\partial}{\partial\theta}\mathcal{C}_{single} = \sum_{i=0}^{1} c_i \mathcal{C}_{single}\left(a_i\theta + s_i\right) \tag{20}$$

Using these components, it was possible to compute the empirical Fisher information matrix. The parameters of the QVR model were initialised randomly 100 times, and for each initialisation, $k = 100$ pairs (x, y) were sampled to calculate an empirical Fisher information matrix. This resulted in 100 different matrices, one for each parameter initialisation, after computation in a simulated environment. The average spectrum is shown in Fig. 12

The more evenly distributed the average spectrum, the better the model capacity and trainability (Abbas et al. 2021). The eigenvalues are plotted on a logarithmic scale, and this graph does not necessarily show an even spread of eigenvalues, but this needs to be judged through comparison to other models attempting the same task. As such, whether or not this approach describes an advantage of QVR remains to be seen.

Aside from providing a basis for comparison for future work, an advantage which can be pointed to empirically is the number of parameters. Classical ML tends to be liberal with the number of trainable parameters in a model because using more parameters allows more complex functions to be approximated. On the other hand, QVR approach has shown promise with a mere 24 parameters. Because the way quantum computers work is fundamentally different from the way classical computers work, it would be unfair to say that QVR is simpler because it uses less parameters. Nevertheless, it is encouraging that only a few parameters need to be learned to be able to capture the information necessary to separate GWs from background noise reasonably well.

10 Discussion

Numerous results have been presented in this work, and this section is concerned with explaining these. Some are clearly explained while for others suggestions are made. In some cases, more investigation is needed.

The convergence patterns during QVR training tend to go through periods of relative stability followed by sharp drops. It is possible that this tendency may be a result of the barren plateau phenomenon, which occurs when loss gradients with respect to quantum circuit parameters tend towards zero with only a small deviations (McClean et al. 2018). The effect of barren plateaus on the results presented is likely limited seeing as the effect is less severe for a small number of qubits. For all of the runs, the loss during training flatlined after initially decreasing. It could be that certain patterns of non-anomalous data were learned at this point, hence the initial decrease, but that others were too complex to pick up on, hence the flatlining before reaching zero. While all models ended up around the same loss value after training, only one of the five which used filtered data performed poorly. The reason for this is unknown.

A potential cause of the spikes in these convergence patterns was the mini-batch size being too small. If difficult training examples made up a large enough portion of a mini-batch, the loss for that mini-batch will have been especially high. Assuming difficult examples made up the minority of the training data, and were therefore less likely to be sampled, the chance of having a disproportionately high number of them would have been smaller for larger mini-batches. Because only 100 data samples were taken per loss calculation, it is possible that only a small number of these contributing an abnormal amount to the loss is already enough to cause a spike. It seems logical to attribute these spikes to the presence of glitches in the training data. The apparent robustness of QVR to the presence of a small proportion of glitch signals, however, contradicts this idea. This is evident by the lack of a clear overlap between training data from observation periods marked as unsuitable for analysis, indicated by red bars in the loss graphs, and the spikes in these graphs. Although filtering likely removed some glitches, there was also no clear overlap for the unfiltered data either. It could be that regular observational periods

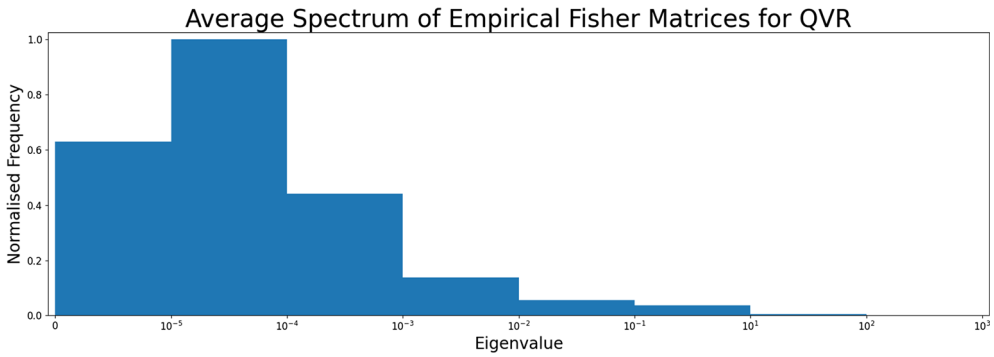

Fig. 12 Average spectrum of empirical Fisher matrices for 100 sets of randomly initialised parameters

produce signals which are difficult enough to cause these spikes. Seeing as the spikes during training did not seem to impact the performance of QVR, no attempt was made to mitigate them with larger mini-batches.

Recall that each model learned a parameter η, which after training acted as a constant to compare to circuit output via Eqs. 12 and 13. Because during training these parameters tended to shift closer to zero, shown in Table 4, it may be that high loss was associated with parameterisations where η was far from zero.

As a proof of concept, QVR for GW detection is promising. Perfect accuracy was achieved with noiseless simulations on filtered GW data on the, albeit limited, testing sets for multiple runs using different starting conditions and different non-anomalous data partitions for training, threshold setting, and testing. The time complexity of filtering the signals dominates that of the QVR algorithm asymptotically, giving rise to a total time complexity of $\mathcal{O}(N \log N)$ for a signal of length N. This is an improvement over MF, which has $\mathcal{O}(MN \log N)$ for M templates. The performance of QVR on unfiltered GW data was worse but still respectable. This decrease in performance may be worth it because leaving out the filtering process removes the time complexity associated with the fast Fourier transform. This leaves the time complexity of QVR, $\mathcal{O}(N)$, as the most expensive component of the GW anomaly detection process described in this work. This upper bound does not take into account the extra time complexity incurred by simulating a quantum algorithm on a classical computer.

In principle, QVR can be run in parallel for an additional speed-up. With access to unlimited resources, the contribution of each $(value, time)$ tuple could be computed by a different quantum computer, with the only remaining step being to average these contributions. Most ML models which deal with time series depend on the sequential or simultaneous processing of values, and thus are not subject to this kind of improvement. In reality, a more modest speedup can be expected due to the scarcity of quantum computers, like the $\times 2$ speed-up achieved with the original implementation of QVR by splitting computations between two quantum computers (Baker et al. 2023). Additional speedup can be achieved by computing anomaly scores by averaging over a subset of the input time series, like was done for the tests on ibm_nairobi. Using as few as 70 from the 4096 tuples each, the anomaly was clearly separated from the non-anomaly without having to adjust the classically trained threshold.

Even if QVR worked perfectly on quantum hardware to achieve the aforementioned time complexity improvements, the classifications made only identify anomalous signals without linking them to specific events. As such, the use of QVR in GW detection would be in part of a search pipeline, in which it would flag interesting signals for further analysis by other means. The usefulness of a component of the search pipeline which can identify promising signals efficiently and effectively should not be underestimated, as in both real-time searches and database scans, computing resources can be a limiting factor. Because of this, QVR is worth considering as an approach.

It was suggested that the success of QVR hinges on whether or not non-anomalous states $\Gamma = \{|\overrightarrow{x_{i,j}}\rangle : 1 \leq i \leq m, 1 \leq j \leq n\}$ can be evolved as $\left|\overrightarrow{x^*_{i,j}}\right\rangle = e^{iHj} |\overrightarrow{x_{i,j}}\rangle$ such that upon measurement, the resulting state $c\left(\overrightarrow{x_{i,j}}\right) = \left\langle \overrightarrow{x^*_{i,j}} \right| \hat{O} \left| \overrightarrow{x^*_{i,j}} \right\rangle$ is approximately the same for each state in Γ. This is the case when the difference between any two of the states after measurement is bounded by some positive ϵ, so $\left|c\left(\overrightarrow{x_{i,j}}\right) - c\left(\overrightarrow{x_{i',j'}}\right)\right| < \epsilon$ (Baker et al. 2023). The ability of QVR to distinguish GW signals from noise signals implies that this condition likely holds for quantum states created by encoding noise signals into a superposition. Here, Γ is the set of states created by angle encoding individual values from noise signals from LIGO Livingston, where i indicates which background noise signal is under consideration and j the specific time in that signal. This condition therefore suggests that evolving these quantum states with the same Hamiltonian H for a time given by j and the measuring with some observable $\hat{O}$ results in expected values which are close to each other.

Based on the performance of the QVR, it seems that the choice of sampling frequency for the interferometers signals was appropriate. There is a tradeoff that comes with increasing sampling frequency. On one hand, more values in an input time series would allow the algorithm to capture behaviour which takes place on short time scales. On the other hand, the goal remains to perform GW searches in a timely manner, and more data to process means QVR will take more time to pick out anomalies. The sampling frequency should be chosen to strike a balance between the two. The data from LIGO Livingstone was originally collected at 16,384 Hz, so the sampling rate can certainly be adjusted.

11 Limitations

While the results reported in this work indicate that QVR has potential in GW detection, some should be seen as first steps rather than as definitive claims. In any case, more testing is required to fully determine the feasibility of QVR for practical use.

For one, more extensive experiments need to be run on NISQ devices. While the results on two signals are encouraging, it would unfair to claim that quantum computers can perform anomaly detection on GW data without testing more signals. Also, the reported quantum processing unit times did not include the latency overhead involved in submitting jobs and extracting them once completed. Moreover, using the full

4096 tuples in a signal to make a classification would increase the reported time. Reducing the signals to a length of 70 to save is an attractive option to save time. However, while it worked on the two tested signals, it is possible that for others, this approach may be detrimental to performance seeing as the contributions to the anomaly score from important parts of the time series could be skipped over.

In principle, QVR should pick up any deviation from the behaviour of noise signals as anomalies. However, here the threshold was tuned so as to maximise performance on known examples of GWs. Because these are all from compact binary coalescence events, no testing of the capabilities of QVR on anomalies from other types of GW was performed. It is possible that the threshold which worked for the GW signals tested in this work does not work as well for other types of GW. Additionally, a total of 78 anomalies were used for testing. Ideally, more real GW signals would be used for testing but these were not available. Instances where perfect accuracy was reported should therefore be taken with a grain of salt.

The general unitary operator in Eq. 9 can be decomposed into the three rotations Z, Y, and Z in that order (Krol et al. 2022). Therefore, this series of gates, which requires only two angles to be learned, could have been used instead of X, Y, and Z, which requires three angles to be learned.

This work assumed that it was possible to discern which signals are from GWs using only one second of those signals. This may not be the case for certain GWs. For example, signals produced by continuous GWs will likely require consideration over a longer time window. Another drawback is that, with each hardware update, the QVR model will need to be retrained to learn to characterise the behaviour of the noise associated with the new interferometer hardware.

An advantage of MF is that it proves a degree of statistical certainty a given signal is associate with a specific GWs. Quantum variational rewinding indicates how far an input signal is from the classification threshold, which can be seen as the model's confidence in a classification, but there is no formal measure of confidence, which is also the case for classical ML approaches (Gebhard et al. 2019). As such, this approach is best combined with existing methods which do not suffer from such drawbacks.

12 Conclusion

Gravitational waves carry information about the distant universe which scientists can learn from. Already, much has been learned the confirmed detections, and Appendix B highlights some discoveries that were possible due to their detection. For this reason, the use of QVR for GW detection is of interest and was investigated in this work.

The QVR algorithm saw varying degrees of success in a noiseless simulated setting when classifying known GW signals as anomalies. Using filtered data, perfect accuracy was achieved, and using unfiltered data, an accuracy of approximately 0.904 was achieved at perfect recall. It was shown that the algorithm runs on quantum hardware, but insufficient results were generated to reach a general conclusion about how well QVR works on a quantum computer.

The advantages over MF offered by QVR are the independence from templates to allow searches for unknown wave forms, asymptotic time complexity improvements, and the possibility to perform GW searches without filtering signals before hand. These features come at a cost. Anomalous signals are identified without saying anything about the event causing the GWs, and without providing a statistical degree of certainty that a signal classified as an anomaly is actually a GW.

Determining whether QVR offers an inherently quantum advantage over classical approaches is difficult. Numerical indications of model capacity and trainability were provided such that future work can be compared to QVR. This was done so that QVR can be shown to be more or less effective than classical ML approaches in these regards.

Not only are new GW detectors under development, but quantum computers are also continually improving. It is probable that the current online nature of quantum computers will be replaced by hybrid quantum-classical setups in high performance computing centres. Even though QVR is well suited to the NISQ era and this exploratory investigation is promising, these developments are likely facilitate data analysis on GWs with sources other than binary black hole coalescences.

13 Future work

As a next step, the feasibility of and QVR for large-scale GW searches should be determined. The focus should be on performance and run time in practice. In addition to this, work should look into the detection of new GWs with QVR. This can be done, for example, by lowering the tunable threshold until signals which are not caused known GWs are classified as anomalies. Investigating these signals could shed light on previously undetected GWs.

Exploring the effectiveness of using signals from multiple sources simultaneously for GW detection is another interesting avenue for further research. The algorithm works with multi-dimensional time series (Baker et al. 2023), and a tradeoff between the additional information from additional detectors and the information lost to decoherence due to a deeper circuit most likely exists. This could be especially relevant moving forwards as detectors are built which produce multiple streams of data (Meijer et al. 2023).

Finally, readers are encouraged to compare the spectrum of the empirical Fisher information matrix of future ML models for GW detection to the one provided here. This could help determine whether QVR is in accordance with evidence that suggests quantum ML models have better capacity and trainability than their classical counterparts (Abbas et al. 2021).

Appendix A: Gravitational wave emission and detection

Although an in-depth treatment of GW emission (Perlick 2023) is outside the scope of this work, a general overview is given here. General relativity dictates that spacetime, consisting of three spatial coordinates and one time coordinate, is a four-dimensional manifold equipped with a metric tensor which defines distances between points. Taking the derivative of the metric tensor produces a geodesic equation describing the curvature of the manifold, and evaluating this at a point gives the forces acting on some particle due to gravity. In this way, curved spacetime is the cause of gravity. Gravitational waves are packets of energy which propagate by perturbing the metric tensor, which constitutes a warping of spacetime.

The original formulation of Einstein's field equations is shown in Eq. 21, in which the Einstein tensor $G_{\mu\nu}$ is related to $T_{\mu\nu}$, the energy-momentum tensor, via the universal gravitational constant G. On the left-hand side, the Einstein tensor is a function of the metric tensor, which describes gravity as a consequence of curved spacetime, and the Ricci tensor, which describes the deviation from flat spacetime as measured by the volume change caused by the deviation of geodesics from straight lines. On the right-hand side, the energy-momentum tensor encodes how different forms of energy are distributed at a point in space. Analogous to how mass density is the cause of a gravitational field in Newtonian gravity, this description of energy is the cause of a gravitational field in general relativity. This relationship describes how energy changes the geometry of space time and vice versa. Einstein's notation of lower and upper indices is used in general relativity which imply summation and not exponentiation. μ and ν indicate that the tensors are of rank two. There are 4 rows and three columns, with one of each being for time entries and the remaining three rows and three columns being for spatial entries.

$$G_{\mu\nu} = 8\pi G T_{\mu\nu} \tag{21}$$

Equation 21 is deceptively concise, and finding exact solutions under this formulation is arduous. Fortunately, a simplifying assumption can be made leading to linearised gravity. This assumption entails that the gravitational field at the point of interest is weak, so spacetime is approximately Minkowski, meaning that GWs can be treated as perturbations on otherwise flat spacetime. On earth, the terms discarded because of this assumption would have had negligible impact if they were kept. The emission of GWs takes place when a gravitational quadrupole moment behaves in a certain way. A gravitational quadrupole moment is a distribution of mass around a centre. It is non-zero when this distribution is non-uniform. When this distribution of mass around a centre changes, the quadrupole moment generates a GW. For example, expanding a sphere would not generate GWs, but squashing it such that it flattens does. Equation 22 is the gravitational quadrupole moment, where T^{00} is the entry of the energy-momentum tensor corresponding to energy density, and Eq. 23 describes how the behaviour of this quadrupole moment is linked to $\gamma_{\mu\nu}$, which is the perturbation of the metric tensor.

$$Q^{kl}(t) = \int_{K_R} T^{00}(t, \vec{v})\, x^k x^l dV \tag{22}$$

$$\frac{2\pi r}{\kappa}\gamma^{kl}(t, \vec{v}) = \frac{1}{2c^2}\frac{d^2}{dt^2}Q^{kl}\left(t - \frac{r}{c}\right) \tag{23}$$

The indices k and l have replaced μ and ν to show that the gravitational quadrupole Q is a composed of spatial information without a time component. In Eq. 22, an integral is taken over a region K_R of the (0, 0) entry of the energy-momentum tensor. Under linearised gravity, the energy-momentum tensor is assumed to be zero outside the region generating GWs so that the only gravitational effect in the far zone, which is the region of interest far from the source, is that of the GWs and not the gravitational field induced by the energy-momentum tensor. As such, it is only necessary to consider the emitting region. The integrand is the energy density at a time t and a position $\vec{v}$ multiplied by two spatial coordinates. In Eq. 23, the perturbation of the spatial components of the metric tensor γ at time t and position $\vec{v}$ is related to the second derivative of the quadrupole moment with respect to time through constants. The argument of the quadrupole moment is the retarded time, which is the difference between the time of the emission, t, and the time that the GW arrives in the far zone, as calculated by the distance travelled and the speed of the GW, $\frac{r}{c}$. Implied by the second time derivative, acceleration of the quadrupole moment is required for GWs to be generated. More specifically, asymmetric acceleration of components making up the quadrupole is required, as this changes the mass distribution. The result is a plane harmonic wave with form dictated by Eq. 24. This is a description of the Euclidean distance by which a particle is offset by a GW, where x and y are positions in an arbitrary co-system and a free-falling coordinate system with respect to the particle in question respectively. A_+ and $A_\times$ are the two polarisation states of the GW, and the presence of sinusoidal terms

indicates that the particle is indeed being offset by a wave. Indices i and j run from one to three to represent spatial coordinates.

$$\delta_{ij} y^i y^j = \delta_{ij} x_i x_j + |A_+| \left(\left(x^1\right)^2 - \left(x^2\right)^2 \right) \cos(\phi - \omega t) + 2 |A_\times| x^1 x^2 \cos(\psi - \omega t) \quad (24)$$

Detecting GW is done with laser interferometers, which take advantage of the fact the GWs propagate by warping spacetime. A laser is split into two components that run down perpendicular tunnels of equal length. At the end of both tunnels hangs a mirror which reflects the beams back so that they meet and interfere with each other before the recombined laser is measured. A passing GW wave will stretch one tunnel while simultaneously compressing the other resulting in a continuous change in the interference pattern while the wave is passing through. Across short time spans with slowly varying amplitude modulation functions this response is modelled by Eq. 25, although other types GW signal require different treatment (Jaranowski and Królak 2012). The response h is a function of time t based on constant amplitude A_0, initial phase ϕ_0, and extrinsic parameters of the GW ζ^u. These extrinsic parameters also define the amplitude modulation function g of t and the phase ϕ of t.

$$h\left(t; A_0, \phi_0, \zeta^u\right) = A_0 g\left(t; \zeta^u\right) \cos\left(\phi\left(t; \zeta^u\right) - \phi_0\right) \quad (25)$$

Appendix B: The relevance of gravitational waves

The first GW detection was also the first observation of a binary black hole merger (Abbott et al. 2016) and allowed for the calculation of various properties of the objects involved like orbital frequencies, orbital separation, velocities, and masses before the merger, as well as properties like mass and spin after it. Subsequent detections of GWs emitted by binary black hole mergers have given astronomers insights into other properties of the sources, like the spin pre-merger (Abbott et al. 2016) and sky location post-merger (Abbott et al. 2017). While GWs can provide information about the objects emitting them in the absence of other evidence like electromagnetic counterparts (Abbott et al. 2020), they also bring forth new opportunities in multi-messenger astronomy in that GWs data can be used in tandem with evidence produced by other astrophysical processes to form a more complete picture of such events (Abbott et al. 2017). Gravitational waves travel at the speed, and because they take time to propagate, information gleaned by their detection allows us to determine what the universe looked like at the time of their emission, with some confirmed detections stemming from events dating back to around two billion years after the big bang (Belczynski et al. 2016). On top of information about GW sources imparted by such detections, they also yield consequences relating to fundamental physics theory. The correspondence between detected GWs and the predictions of not only their existence but also their strength, supplements the body of evidence for Einstein's theory of general relativity while ruling out alternative theories of gravity (Pardo et al. 2018; Boran et al. 2018).

Because this work deals with both general relativity and quantum mechanics, the question of whether there is a related quantum gravity arises. There are two distinct phases involved in detecting GWs with QVR, the translation of GW waves into a time series and the processing of the time series with a quantum computer. Quantum variational rewinding is absent from the first phase and the general relativistic description of general relativity is absent from the second. In this way, this work does not involve an overlap between quantum computing and general relativity and is unrelated to quantum gravity.

Appendix C: Mathematical details of matched filtering

The concepts in MF covered here are a skeleton of more extensive overviews (Turin 1960; Janquart 2020). Consider the signal produced by a detector as a starting point. It can be written as an additive combination of a GW component $h(t)$ and a noise component $n(t)$ at time t, where both components are individually zero centred. When there is a GW traversing through the detector, the signal is $i(t) = n(t) + h(t)$ as opposed to $i(t) = n(t)$ in the absence of a GW. It is this signal that is used at the input to a matched filter, which is a linear transformation that produces an output signal $o(t)$. Because the system is linear, its response to an impulse is the convolution of an impulse response and the impulse itself. In this case, then, the output is the convolution $o(t) = k(t) * i(t)$ of the impulse response function $k(t)$ and the input $i(t)$. Converting to the frequency domain allows this to be formulated as $z(\omega) = \mathcal{K}(\omega)\mathcal{I}(\omega)$ where ω is angular frequency. Taking the inverse Fourier transform then leaves Eq. 26.

$$o(t) = \int_{-\infty}^{\infty} \mathcal{K}(\omega)\mathcal{I}(\omega) e^{i\omega t} d\omega \quad (26)$$

Because it is assumed that a template of the sought-after wave $h(t)$ is available, $k(t)$ can be designed as $k(t) = ak(T - t)$, which is the template scaled by a constant a, delayed by a constant T, and reversed. A consequence of this is that the transfer function $\mathcal{K}(\omega)$ from Eq. 26 takes the form given by Eq. 27 for some t. The numerator is the Fourier-

transformed impulse response function, and the denominator is the full power spectral density of the input signal. The latter is the Fourier transform of the input signal's autocorrelation function.

$$\mathcal{K}(\omega) = \frac{a\mathcal{I}^*(\omega)\, e^{-i\omega t}}{|\mathcal{N}(\omega)|^2} \tag{27}$$

Now it is possible to define the signal to noise ratio captured by Eq. 28. The numerator is the square of output from Eq. 26, which is the power of the template in the input signal, and the denominator is inverse Fourier transformed square of the transfer function, which is the power spectral density of the noise component of the input signal. Importantly, a function has been obtained which describes of how much of the input signal is composed of a template signal and how much is made up of noise. Times for which this ratio is high correspond to when the presence of the template is high. The difference between the measured signal-to-noise ratio and that of the expected value when the template GW is present can be used to perform a χ^2 test.

$$\rho(t) = \frac{2\left(\int_{-\infty}^{\infty} \mathcal{K}(\omega)\, \mathcal{I}(\omega)\, e^{i\omega t} d\omega\right)^2}{\int_{-\infty}^{\infty} (\mathcal{K}(\omega))^2\, d\omega} \tag{28}$$

By making use of the Cauchy-Schwarz inequality, it can be shown that the signal-to-noise ratio is maximised when the impulse response is a scaled, delayed, and reversed version of the template, and it is from this matching of the filter and the sought-after signal that MF gets its name. Matched filtering makes the assumption that the background noise is both stationary and Gaussian. Unfortunately, the background noise from laser interferometers does not always adhere to this assumption due to the workings of certain components of the detector and terrestrial factors. Such noise can often be either mitigated or marked as part of a segment of data which unsuitable for analysis (Abbott et al. 2020). Another cause for deviation from these assumptions are incoming stochastic GWs, which are combinations of weak GWs coming in from all directions and appear as additional background noise (Romano and Cornish 2017).

Appendix D: Relation to the no fast-forwarding theorem

The no fast-forwarding theorem states that simulating quantum evolution is impossible with a quantum gate complexity which is less than the evolution time up to some predefined maximum evolution time (Gu et al. 2021). When the Hamiltonian driving the evolution is completely unknown, this holds, but arbitrary violations are possible when the Hamiltonian is completely known (Atia and Aharonov 2017). Research in fast-forwarding quantum evolution deals with violations caused by classes of Hamiltonians which are only partially known.

A general procedure for simulating Hamiltonian evolution variationally when no assumption is made about its structure can lead to violations of the no fast-forwarding theorem, but not arbitrarily. That is, knowing the actual entries of the Hamiltonian does not necessarily mean that a general procedure for simulating the evolution it brings about arbitrarily violates the no fast-forwarding theorem. This is because the conversion to a gate-based representation induces a tradeoff between gate complexity and simulation accuracy. The direct learning of the parameters of a gate-based representation of a target Hamiltonian in QVR constitutes a full knowledge of the Hamiltonian, and thus allows for arbitrary violations of the no fast-forwarding theorem. This is because it avoids the conversion to a gate-based representation.

Author contribution B.R.M performed the research and wrote the manuscript under the supervision of M.D., N.N., and W.S. All authors reviewed the manuscript.

Data availability All code, with the QVR implementation being heavily based on the original implementation by Baker et al. (2023), has been made publicly available (Github Repository 2023).

Declarations

Conflict of interest The authors declare no competing interests.

References

Aasi J et al (2015) Advanced LIGO. classical quant grav 32(7). https://doi.org/10.1088/0264-9381/32/7/074001

Abbas A et al (2021) The power of quantum neural networks. Nat Comput Sci 1(6). https://doi.org/10.1038/s43588-021-00084-1

Abbott BP, et al (2016) GW151226: observation of gravitational waves from a 22- solar-mass binary black hole coalescence. Phys Rev Lett 116(24). https://doi.org/10.1103/PhysRevLett.116.241103

Abbott BP, et al (2016) Observing gravitational-wave transient GW150914 with minimal assumptions. Phys Rev D 93(12). https://doi.org/10.1103/physrevd.93.122004 ISSN: 2470-0029

Abbott BP, et al (2017) GW170814: a three-detector observation of gravitational waves from a binary black hole coalescence.

Phys Rev Lett 119(14). https://doi.org/10.1103/PhysRevLett.119.141101
Abbott BP et al (2019) Searches for continuous gravitationalwaves from 15 supernova remnants and fomalhaut b with advanced LIGO. Astrophys J 875(2). https://doi.org/10.3847/1538-4357/ab113b
Abbott BP et al (2016) Observation of gravitational waves from a binary black hole merger. Phys Rev Lett 116(6). https://doi.org/10.1103/PhysRevLett.116.061102
Abbott BP et al (2017) GW170817: observation of gravitational waves from a binary neutron star inspiral. Phys Rev Lett 119(16). https://doi.org/10.1103/PhysRevLett.119.161101
Abbott BP et al (2017) Multi-messenger observations of a binary neutron star merger. Astrophys J Lett 848(2). https://doi.org/10.3847/2041-8213/aa91c9
Abbott BP et al (2020) A guide to LIGO-Virgo detector noise and extraction of transient gravitational-wave signals. Classical Quant Grav 37(5). https://doi.org/10.1088/1361-6382/ab685e
Abbott R et al (2021) GWTC-3: compact binary coalescences observed by LIGO and virgo during the second part of the third observing run. Phys Rev X 11(6). https://doi.org/10.48550/arXiv.2111.03606
Abbott R et al (2023) Open data from the third observing Run of LIGO, Virgo, KAGRA, and GEO. Astrophys J Suppl Ser 267(2). https://doi.org/10.3847/1538-4365/acdc9f
Abbott R, et al (2020) GW190814: gravitational waves from the coalescence of a 23 solar mass black hole with a 2.6 solar mass compact object. Astrophys J Lett 896(2). https://doi.org/10.3847/2041-8213/ab960f
Atia Y, Aharonov D (2017) Fast-forwarding of Hamiltonians and exponentially precise measurements. Nat Commun 8. https://doi.org/10.1038/s41467-017-01637-7
Baker JS, et al (2023) Quantum variational rewinding for time series anomaly detection
Baker PT et al (2015) Multivariate classification with random forests for gravitational wave searches of black hole binary coalescence. Phys Rev D 91(6). https://doi.org/10.48550/arXiv.1412.6479
Belczynski K et al (2016) The first gravitational-wave source from the isolated evolution of two stars in the 40–100 solar mass range. Nature 534(7608). https://doi.org/10.1038/nature18322
Bergholm V, et al (2022) PennyLane: automatic differentiation of hybrid quantumclassical computations. https://doi.org/10.48550/arXiv.1811.04968
Boran S et al (2018) GW170817 falsifies dark matter emulators. Phys Rev D 97(4). https://doi.org/10.1103/PhysRevD.97.041501
Cabero M et al (2019) Blip glitches in advanced LIGO data. Classical Quant Grav 36(15). https://doi.org/10.1088/1361-6382/ab2e14
Cannon K et al (2010) Singular value decomposition applied to compact binary coalescence gravitational-wave signals. Phys Rev D 82(4). https://doi.org/10.1103/physrevd.82.044025. ISSN: 1550-2368
Christ M et al (2018) Time series feature extraction on basis of scalable hypothesis tests (tsfresh-a Python package). Neurocomputing 307(1). https://doi.org/10.1016/j.neucom.2018.03.067
L. Chuang I, Nielsen MA (2010) Quantum computation and quantum information. In: Cambridge University Press. Chap. 7.1 Guiding Principals
Cîrstoiu C et al (2020) Variational fast forwarding for quantum simulation beyond the coherence time. NPJ Quant Inf 6. https://doi.org/10.1038/s41534-020-00302-0
Cooley JW, Lewis PAW, Welch PD (1967) Historical notes on the fast Fourier transform. Proc IEEE 55(10). https://doi.org/10.1109/PROC.1967.5959
Couvares P, et al (2021) Gravitational wave data analysis: computing challenges in the 3G era. Report by the GWIC 3G data analysis computing challenges subcommittee. https://doi.org/10.48550/arXiv.2111.06987
Cresser JD (2011) Quantum physics notes. Macquarie University. Chap. 15.1 Stationary States, 15.2 The Schrodinger Equation–a 'Derivation'
Cuoco E et al (2020) Enhancing gravitational-wave science with machine learning. Mach Learn Sci Technol 2(1). https://doi.org/10.1088/2632-2153/abb93a
Davis D et al (2021) LIGO detector characterization in the second and third observing runs. Class Quant Grav 38(13). https://doi.org/10.1088/1361-6382/abfd85
Dreissigacker C et al (2019) Deep-learning continuous gravitational waves. Phys Rev D 1(4). https://doi.org/10.1103/PhysRevD.102.022005
Gabbard H, Williams M, Hayes F, Messenger C (2018) Matching matches filtering with deep networks for gravitational-wave stronomy. Phys Rev Lett 120(14). https://doi.org/10.1103/PhysRevLett.120.141103
Gao S et al (2022) A quantum algorithm for gravitational-wave matched filtering. Phys Rev Res 4(2). https://doi.org/10.1103/PhysRevResearch.4.023006
Gebhard TD et al (2019) Convolutional neural networks: a magic bullet for gravitational-wave detection? Phys Rev D 100(6). https://doi.org/10.1103/physrevd.100.063015
Github Repository (2023) https://github.com/brodriguesdemiranda/QVR4GW
Gobeil J (2023) GWQML. Python Package. https://gitlab.com/jgobeil/gwqml Accessed 24-Oct-23
Gu S, Somma RD, Şahinoğlu B (2021) Fast-forwarding quantum evolution. Quantum 5(1). https://doi.org/10.22331/q-2021-11-15-577
Janquart J (2020) Gravitational waves signal analysis: matched filtering, typical analyses and beyond. Master Thesis. Liége Université
Jaranowski P, Królak A (2012) Gravitational-wave data analysis formalism and sample applications: the Gaussian case. Living Rev Relativ 15(1). https://doi.org/10.12942/lrr-2012-4
Kim K, Harry IW, Hodge KA, Kim YM, Lee CH, Lee HK, Oh JJ, Oh SH, Son EJ (2015) Application of artificial neural network to search for gravitational-wave signals associated with short gamma-ray bursts. Classical Quant Grav 32(24). https://iopscience.iop.org/article/10.1088/0264-9381/32/24/245002
Klimenko S et al (2016) Method for detection and reconstruction of gravitational wave transients with networks of advanced detectors. Phys Rev D 93(4). https://doi.org/10.1103/physrevd.93.042004
Krol AM et al (2022) Efficient decomposition of unitary matrices in quantum circuit compilers. Appl Sci 12(1). https://doi.org/10.3390/app12020759
Liao H, et al (2023) Machine learning for practical quantum error mitigation
Macleod DM et al (2021) GWpy: a python package for gravitational-wave astrophysics. SoftwareX 13. https://doi.org/10.1016/j.softx.2021.100657
McClean JR et al (2018) Barren plateaus in quantum neural network training landscapes. Nat Commun 9(1). https://doi.org/10.1038/s41467-018-07090-4
Meijer Q, et al (2023) Gravitational-wave searches for cosmic string cusps in Einstein telescope data using deep learning. https://doi.org/10.48550/arXiv.2308.12323
Morawski F et al (2021) Anomaly detection in gravitational waves data using convolutional autoencoders. Mach Learn Sci Technol 2(4). https://doi.org/10.1088/2632-2153/abf3d0
Necula V, Klimenko S, Mitselmakher G (2012) Transient analysis with fast Wilson-Daubechies time-frequency transform. J Phys Conf Ser 363. https://doi.org/10.1088/1742-6596/363/1/012032
Nitz A et al (2023) PyCBC. Python Package. https://doi.org/10.5281/zenodo.7746324 Accessed 03-Apr-2023
Pardo K et al (2018) Limits on the number of spacetime dimensions from GW170817. J Cosmol Astropart Phys 2018(7). https://doi.org/10.48550/arXiv.1801.08160

Perlick V (2020) Gravitational waves. Lecture. https://www.youtube.com/watch?v=0CXsCZTLHm0&list=PLaNkJORnlhZk88BPF2MZBEPMasV7m9Qto Accessed 28-Aug-2023

Powell MJD (1964) An efficient method for finding the minimum of a function of several variables without calculating derivatives. Comput J 7(2). https://doi.org/10.1093/comjnl/7.2.155

Qiskit contributors, (2023) Qiskit: an open-source framework for quantum computing. Python Package. https://doi.org/10.5281/zenodo.2573505

Romano JD, Cornish NJ (2017) Detection methods for stochastic gravitationalwave backgrounds: a unified treatment. Living Rev Relativ 20(1). https://doi.org/10.1007/s41114-017-0004-1

Schuld M et al (2020) Circuit-centric quantum classifiers. Phys Rev A 101(3). https://doi.org/10.1103/physreva.101.032308

Schuld M, Petruccione F (2021) Machine learning with quantum computers. Springer Cham, Chap. 4.4.1 Hamiltonian Simulation, 1.1.1 Merging Two Disciplines, 5.3.1 Gradients of Quantum Computations, 5.2.2 An Example: The Pauli-Rotation Encoding

The LIGO Scientific Collaboration et al (2023) Search for eccentric black hole coalescences during the third observing run of LIGO and Virgo. https://doi.org/10.48550/arXiv.2308.03822arXiv:2308.03822. [astro-ph.HE]

Turin G (1960) An introduction to matched filters. Trans Inf Theory 6(3). https://doi.org/10.1109/TIT.1960.1057571

Welch J et al (2014) Efficient quantum circuits for diagonal unitaries without ancillas. New J Phys 16(3). https://doi.org/10.48550/arXiv.1306.3991