Searching for Black Holes Using Auto Differentiation

William Doyle[1],*

[1] University of Portsmouth, School of Mathematics and Physics, Portsmouth, PO1 3FX, United Kingdom

*Corresponding author will.doyle30@gmail.com

Abstract – This study presents GravAD, a novel approach for detecting gravitational waves using automatic differentiation and JAX. GravAD demonstrates comparable signal-to-noise ratio and mass values to established LIGO pipelines with a significant reduction in the number of templates. Limitations include the inability to handle binary neutron star systems and some lower-mass black holes. Leveraging JAX's acceleration, GravAD offers potential as a rapid preliminary tool for gravitational wave detection. Future work includes further optimization of functions, exploration of alternative optimization algorithms, real-time data analysis adaptation, and expanding the scope to handle a broader range of astrophysical sources.

Introduction

Compact binary coalescences (CBCs) are astronomical events that result from the coalescence, or merging, of two distinct compact objects, such as black holes (BHs) or neutron stars (NSs), that form a binary system [1]. Studying BHs and gravitational waves (GWs) emitted during these events has become essential to modern astrophysics. BHs, first proposed by John Michell in 1783 [2], allow scientists to test the laws of physics in extreme environments; while GWs, predicted by Albert Einstein in 1916, enable the observation of previously inaccessible events, such as black hole mergers [3]. The detection of the first GW signal by the Laser Interferometer Gravitational-wave Observatory (LIGO) in 2015 marked a turning point in astrophysics, allowing for a deeper study of the Universe and its constituents [4].

The sequence of a CBC event is characterized by a three-part structure in the waveform pattern, which includes the in-spiral-merger-ringdown (IMR) phases that define the stages of the two interacting objects. This sequence results in a significant release of energy in the form of GWs [1]. These GWs, alternatively ripples or perturbations in space-time, are propagated through space at the speed of light [3]. These ripples in space-time carry information about the properties and dynamics of the bodies that formed them, such as the mass and spins of each compact object, allowing us to verify the existence of binary BHs, which were previously only theoretical predictions [5]. In general relativity, we can approximate a GW, far from the emission source, as a time-dependent perturbation of the space-time metric, allowing us to decipher the information encoded in these GWs by expressing the signal as a pair of dimensionless strain polarizations, h_s and h_c [6]. These data signals are detected in ground-based observatories such as the LIGO Hanford detector in Washington [4].

The LIGO detectors are based on the Michelson interferometer design (Figure 1) and were fundamentally developed by Weiss in the 1970s [7]. LIGO was designed to detect faint signals produced by events billions of years ago [4]. The waveforms relating to CBCs are known as transient-modelled waveforms. These waveforms are characterized by their short duration and rapidly changing amplitude and frequency, reflecting the dynamics of the merging compact binary systems [8] (Figure 2). The waveforms that LIGO collects from its observations allude

to signs of CBCs. To verify a detection, it must first be seen that a captured signal contains a merging binary system. We use a waveform template generated by numerical simulations such as IMRPhenom [9] or effective-one-body (EOB) [10], which output a waveform based on general relativity calculations [11]. This template replicates the signal produced by CBCs, according to the input parameters describing the system. Using a waveform template is essential for verifying a detection as it allows for a comparison between the observed signal with the theoretical GW template. A strong match between the two will result in a high signal-to-noise ratio (SNR), indicating a possible detection [12]. The SNR measures the strength of the GW signal relative to the background noise. It is a critical parameter used in determining the significance of a detection. Due to the dependence of the waveform template on input parameters, it is necessary to generate a template bank that encompasses a myriad of possibilities. This bank can include up to a million waveform templates, allowing for a more detailed data analysis [13]. However, using a larger template bank increases the computational demands for generating and analysing each template, resulting in longer processing times. Hence, there is a trade-off between the level of precision and the computational resources required for the analysis [14].
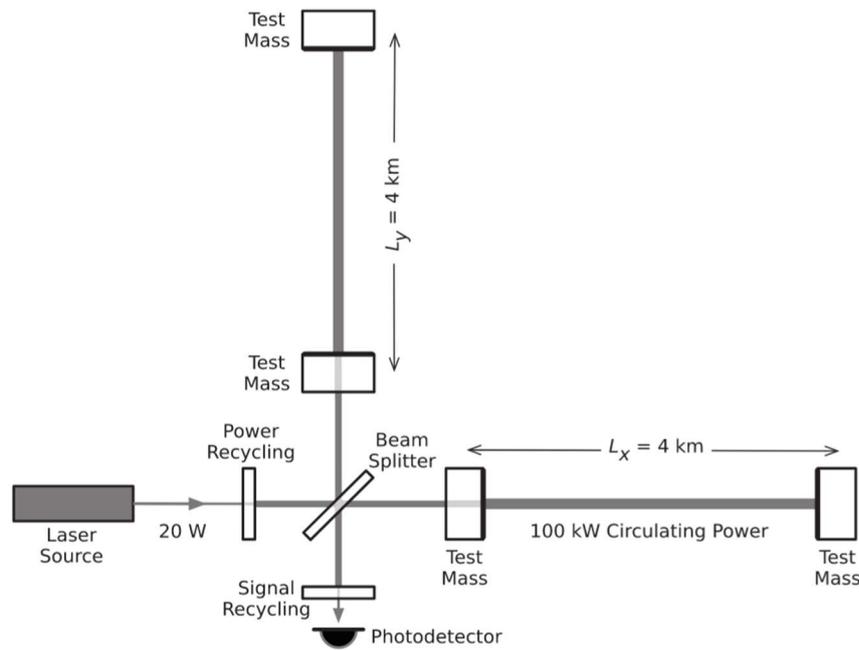
Figure 1. A simplified version of an advanced LIGO detector, a highly sensitive instrument used to detect GWs. If a GW travels perpendicular to the detector plane and is aligned with the 4 km optical cavities, one arm of the detector will get longer. In contrast, the other arm will get shorter during half of the wave cycle [4].

The growing complexity of waveform templates used for GW detections, bolstered by the rapid improvements in the sensitivity of LIGO detectors and, as a consequence, the rising number of GW detections [15], underscores the imperative for the development of effective and resilient data analysis methods for GWs [16]. Furthermore, the emergence of new detectors capable of observing new CBC events necessitates an even more comprehensive analytical approach [17]. Addressing the computational challenges, such as the intensive analysis process associated with large template banks. Therefore, the primary objective of this research project is to develop an innovative approach, is crucial for furthering our understanding of BHs, NSs, and gravity, as well as rigorously testing Einstein's Theory of General Relativity to GW detection.

This research project has two main objectives: to develop GravAD, a custom search pipeline written in Python and compatible with JAX, as a novel approach to GW detection, and to benchmark it against other search pipelines. GravAD uses automatic differentiation (AD), a computational technique capable of evaluating derivatives of functions [18], to dynamically generate and refine waveform templates. These templates are designed to improve their fit to the incoming data with each new iteration, which enhances the efficiency of the detection process. As a consequence of this methodology, we can reduce the number of templates required for data analysis. GravAD can circumvent the need for a brute force approach, which involves analyzing a large number of pre-computed templates. This innovative approach increases the efficiency of the analysis without compromising the quality of the results.
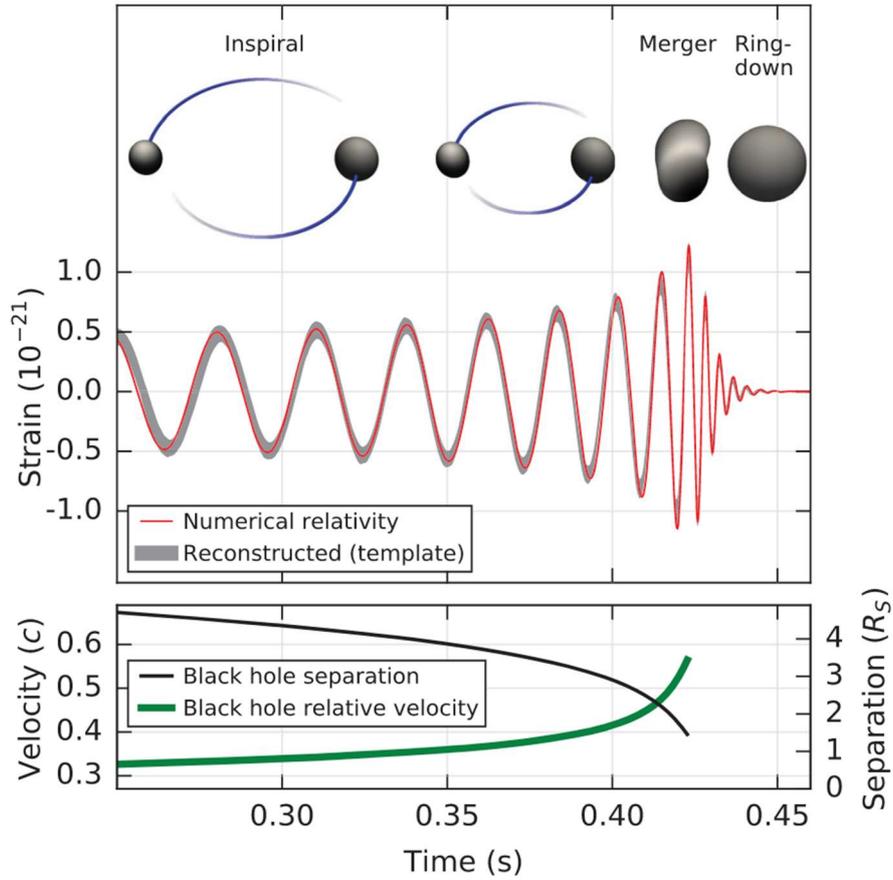
Figure 2. The dynamics of the binary merger and the associated waveform produced by the event [4].

Methodology

In this section, we will explore using AD, matched filtering (MF), and optimization techniques to generate waveform templates for GW data analysis. MF is crucial for detecting weak signals, such as GWs, amidst noisy data. It allows us to find a peak in the SNR, which indicates the presence of a GW and allows us to determine the best-fitting template for the observed strain data.

The optimization techniques used in this project, namely gradient descent, simulated annealing and just-in-time (JIT) compilation, enable us to find the optimal waveform parameters that maximize the SNR. These techniques offer several advantages over traditional methods, such as reduced computational time and generating waveform templates that closely match the observed data. Together, these methods form a novel approach to GW data analysis, potentially improving the search pipeline used to detect the presence of a GW. The delivery of this method is through our pipeline GravAD (Figure 3).
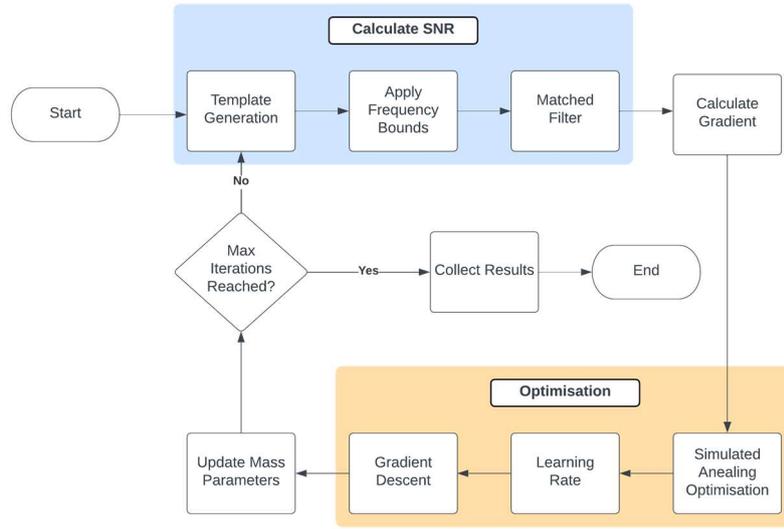
Figure 3. A simplified flow diagram illustrating the sequential operations conducted by the GravAD algorithm. The process initiates with the generation of the first template, followed by the application of frequency bounds to constrain the search to desired frequencies. A matched filter operation is executed with the produced template and the chosen GW signal. Subsequently, GravAD calculates the SNR, and the derivative of this SNR function is taken with respect to each mass value utilized in the template construction. Optimization techniques are then applied, which yield new mass parameters that more accurately fit the data. The algorithm persists in improving data fitting through the iterative refinement of templates until the predefined iteration limit is attained. It should be noted that GravAD is developed using JAX-compatible code, enabling rapid acceleration of this process on GPUs.

Overview of Matched Filtering

Matched filtering is a technique which applies a linear filter to isolate weak signals submerged within stationary and Gaussian noise - the former signifies that the signal properties do not alter over time, while the latter signifies that the noise follows a normal distribution. This methodology effectively maximizes the SNR [19]. In the context of GW detection, MF plays a pivotal role in discerning the template that most accurately aligns with the strain data signal [20]. GW signals, represented as h , are typically contaminated by noise, denoted by n , rendering the observed signal as a complex composite of the signal and noise, expressed as $s = h + n$. In order to find the template that best mirrors the observed signal, it is crucial to utilize a variety of templates that span a comprehensive parameter space. These templates are designed to reflect the unique characteristics of these phenomena [21]. Detectors like LIGO Livingston are employed to record subtle alterations in the detector's arm length and to gather strain data. This data encapsulates information regarding the amplitude, phase, and arrival time of the GW, properties masked by the interference of noise [16]. Further analysis of the events requires signal processing techniques to extract this critical information. The key to extracting this data lies in finding the optimal filter capable of effectively separating the signal from the noise [22]. This optimal filter can be identified by creating a template that successfully mirrors the concealed signal. Nonetheless, the task of analyzing noise can prove demanding due to its non-Gaussian and non-stationary nature, both of which could precipitate false alarms in detection. As such, it becomes imperative to consider statistical tools such as the chi-squared test and coincidence test, which are routinely employed by pipelines like PyCBC [13]. Owing to the massive volume of data that these pipelines process, particularly those dealing with live data, it is essential to recognize the significance of efficient algorithms tailored for large-scale data analysis. In the next part of this section, we delve into the mathematical principles involved in analyzing GWs and the equations our algorithm implements.

The concepts behind MF are tethered to the Fourier Transform (FT), a critical mathematical tool that describes the relationship between a continuous complex-valued function of time, $f(t)$, and its FT, $F(\nu)$, which is a frequency function, given by:

$$F(\nu) = \int_{-\infty}^{\infty} f(t)e^{-j2\pi\nu t} \, dt \qquad (1)$$

Inversely, the function $f(t)$ can be derived from $F(\nu)$ as shown below [23]:

$$f(t) = \int_{-\infty}^{\infty} F(v)e^{j2\pi vt}\, dv \tag{2}$$

The real and imaginary components of $F(v)$ represent the amplitudes and phase constituents of the frequency data inherent in $f(t)$, thus providing another perspective of the signal and, in the context of GWs, allowing us to see the IMR phases more clearly (Figure 4).
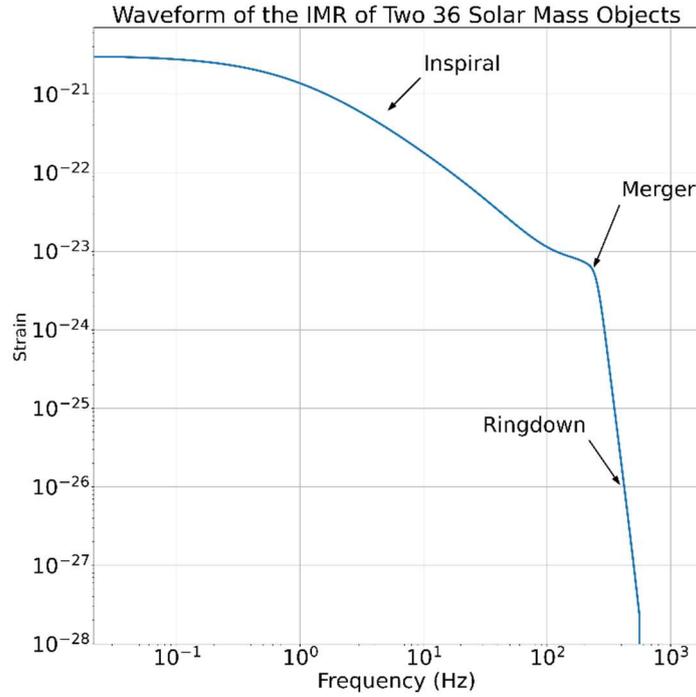

Figure 4. The frequency evolution of the IMR of two 36 solar mass BHs.

Beyond the FT, another pivotal concept integral to MF is the correlation theorem. Correlation plays a fundamental role in MF, establishing a connection between the time and frequency domains. In the time domain, correlation corresponds to multiplication in the frequency domain and vice versa. We can express the correlation between two functions, namely $h(t)$ having a FT of $\tilde{h}(f)$ and $s(t)$ with a FT of $\tilde{s}(f)$, as [23]:

$$\int_{-\infty}^{\infty} h(\tau)s(t+\tau)d\tau \leftrightarrow \tilde{h}(f)\tilde{s}^*(f) \tag{3}$$

This relationship provides a crucial mathematical link in understanding and implementing a matched filter in signal processing and analysis. It allows us to compute the correlation between the template waveform and the data in the frequency domain, providing a quantitative measure of similarity.

Matched filtering is employed to detect well-modelled target signals within detector noise by optimizing the template $h(t)$ across two distinct phases, namely the cosine chirp h_c and the sine chirp h_s as defined by [24]:

$$h_c(t) = \frac{2}{c^2} \left(\frac{\mu}{M_\odot} \right) [\pi GM f(t)]^{\frac{2}{3}} \cos[2\phi(t) - 2\phi_0] \tag{4}$$

$$h_s(t) = \frac{2}{c^2} \left(\frac{\mu}{M_\odot} \right) [\pi GM f(t)]^{\frac{2}{3}} \sin[2\phi(t) - 2\phi_0] \tag{5}$$

The match can be computed at time $t = t_0$ using the complex conjugate of the FT of the template $\widetilde{h}^*$ and the FT of the data es through the matched filtering technique. The calculation is given by [11]:

$$z(t_0) = 4 \int_0^\infty \frac{\tilde{s}(f)\widetilde{h}^*(f)}{S_n(f)} e^{2\pi i f t_0} \, df. \tag{6}$$

$S_n(f)$ is the one-sided Power Spectral Density (PSD) of the detector noise and is used to characterize the noise present at different frequencies [16] (Figure 5), defined by [13]:

$$\langle \tilde{s}(f)\tilde{s}(f') \rangle = \frac{1}{2} S_n(f)\delta(f - f'). \tag{7}$$

The angle brackets denote averaging over noise realizations, and δ is the Dirac delta function. The normalization constant σ that measures the amount of noise in the detector given by [24]:

$$\sigma^2 = 2 \int_{-\infty}^\infty df \frac{\widetilde{h_c}^*(f)\widetilde{h_c}(f)}{S_h(f)} = 2 \int_{-\infty}^\infty \frac{\widetilde{h_s}^*(f)\widetilde{h_s}(f)}{S_h(f)}. \tag{8}$$

This culminates into the test statistic ρ, which is the SNR, where a higher SNR value indicates the presence of a strong match between the theoretical GW waveform template and the data, provided by [11]:

$$\rho(t) = \frac{|z(t)|}{\sigma}. \tag{9}$$
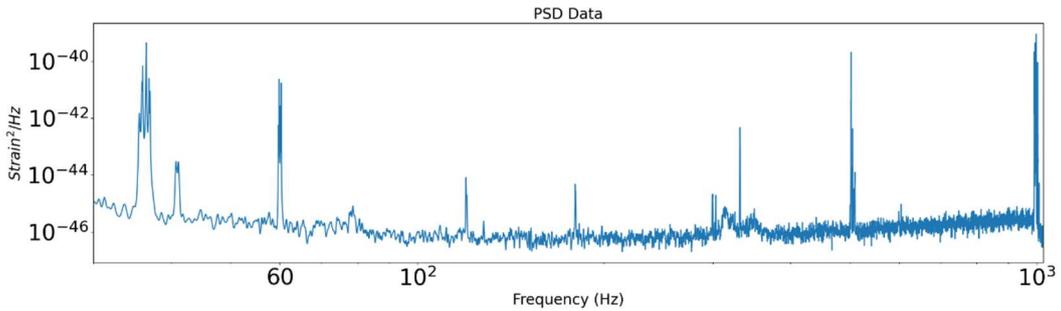
Figure 5. The PSD of noise in a detector where peaks indicate a high power in the corresponding frequency. For instance, 60 Hz is the US mains frequency in the data. The graph was generated using PyCBC [12].

Evaluating the SNR

The SNR is an essential metric regarding the potency of a signal obscured by noise. We can evaluate a template and thus find the parameters of the merger by performing a matched filter. When a template fits well against the data signal, an SNR time-series will display a prominent spike where the two-waveform match. By comparing which template yields the highest SNR value, we can determine which parameters better characterize the data signal (Figure 6). It is essential to recognize that signals which imitate GWs can create a high-value SNR, which with our algorithm, may get detected. Thus, it is vital to recognize techniques which could reconcile this issue, such as false-alarm rate and coincidence tests [25].

An important aspect of evaluating the SNR is the selection of an appropriate threshold to determine a true GW from noise. PyCBC, for the first LIGO observation run, selected a threshold SNR of 5.5 and GstLAL with a threshold of 4 [26]. An optimal SNR threshold is crucial for differentiating signal from noise. Balancing valid GW signal detection and false positive minimization hinges on the threshold choice. A higher threshold offers fewer false positives but may overlook weaker signals, while a lower threshold increases detection sensitivity but risks a higher amount of false positives due to noise fluctuations. The SNR threshold selection depends on the GW search's objectives and detector noise characteristics. Selecting an appropriate SNR threshold is paramount to the detection process; however, it is out of this project's scope to determine the threshold for the SNR. As such, we shall assess the success of the GravAD algorithm by detecting a GW and comparing our SNR value to those obtained by other LIGO pipelines, ensuring the consistency and validity of our results.

a). Representation of a well-fitted template

(b). Representation of a poor-fitted template
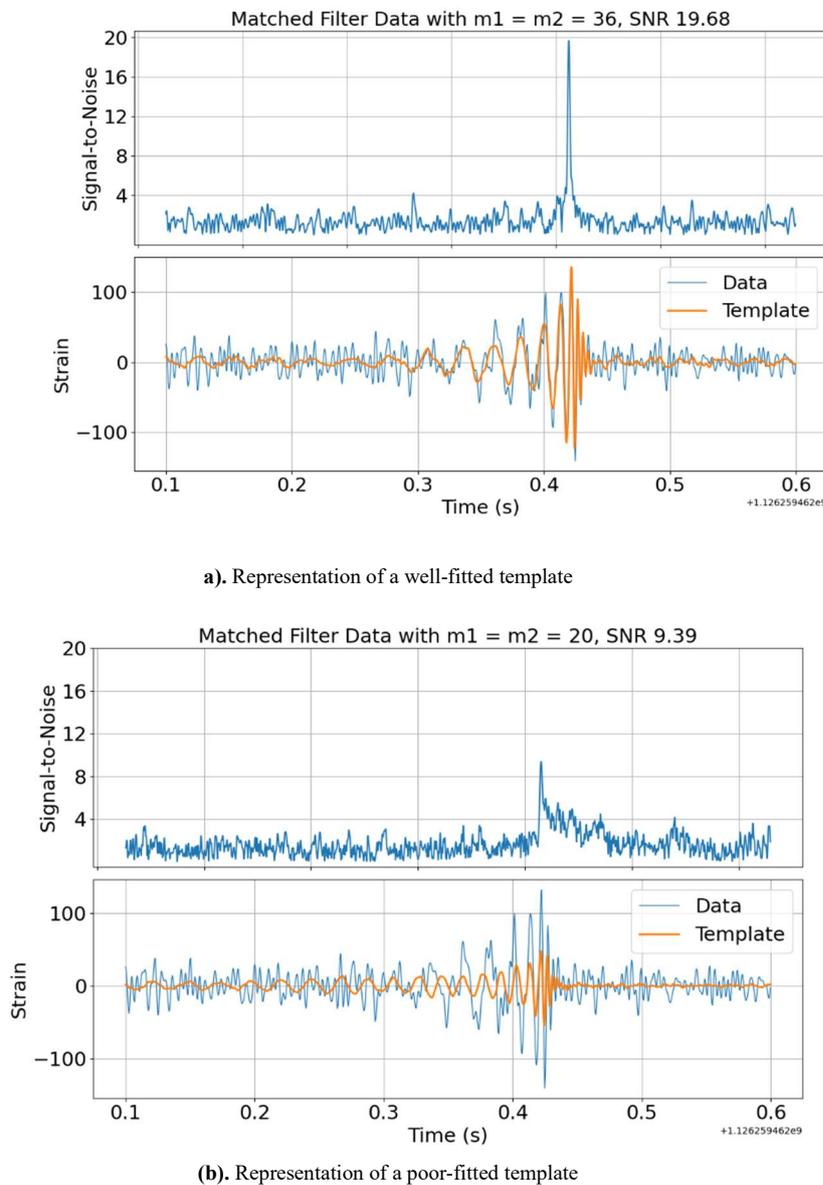
Figure 6. Comparison of SNR between a well-fitted template (a) and a poor-fitted template (b) against real GW data. The well-fitted template shows a prominent peak in the SNR, while the poor-fitted template exhibits a weak peak. The figures were generated using PyCBC [12].

Justification for the Use of Real Data

The decision to incorporate real data into the GravAD pipeline, rather than relying on synthetic or simulated data, is fundamentally underpinned by the principles of validity and reliability. Utilizing real data offers an authentic context that closely parallels the practical applications of our system, consequently enhancing the validity and reproducibility of our findings.

Actual data, derived from genuine GW events as catalogued in the pycbc.catalog, embodies intricate features and attributes that are only present in real signals. By implementing real data, we ensure that the GravAD pipeline is subjected to the comprehensive intricacies of GW signals, thus improving its ability to detect other real CBCs.

Furthermore, the employment of real data in our analysis facilitates the benchmarking of GravAD's performance against LIGO's established standards of pipelines. This comparative approach is essential for evaluating the reliability and robustness of our system, as it enables a direct comparison of our findings with those obtained by fellow researchers utilising identical datasets. This, in turn, bolsters the credibility of our results, thereby demonstrating the potential for GravAD as an alternative to traditional methods.

Data Preprocessing

Before analyzing the GW strain data, we performed several preprocessing steps to clean, normalize, and prepare the data for further analysis. The strain data was first loaded for the event of interest (e.g. GW150914) and then scaled by a factor of 10^{22} to bring it to a suitable range for processing - by avoiding errors relating to issues of precision [27]. The data was then high-pass filtered with a cutoff frequency of 15 Hz to remove low-frequency noise. Following this, the data was resampled to a sampling rate of 2048 Hz to reduce the computational load without losing relevant information [21]. Reducing the noise in the signal is crucial as this will help identify any present signals.

After resampling, the data was cropped to remove filter artefacts, specifically by removing 2 seconds from the start and the end of the time-series. The cleaned strain data was then transformed into the frequency domain using a Fast Fourier Transform (FFT) to facilitate subsequent MF and SNR calculations. The conditioned strain data's PSD was calculated using a 4-second averaging window. The PSD was then interpolated to match the frequency resolution of the conditioned strain data. The resulting PSD was truncated using an inverse spectrum truncation with a low-frequency cutoff of 15 Hz. This part of the algorithm closely followed the PyCBC tutorials [12]. The PSD is an integral part of MF because it reduces the noise profile of certain sources (Figure 7).
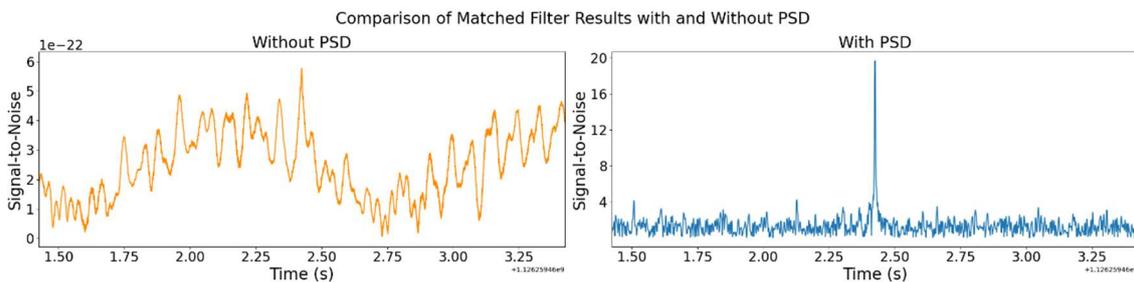
Figure 7. A comparison demonstrating the importance of the PSD for MF. The left graph was matched filtered without a PSD, whereas the right graph was matched filtered with a PSD. The graph was generated using PyCBC [12].

Adjusting the Frequency Series for Seamless Template Generation

A critical step in analyzing GW data involves the comparison of templates with the observed data, which is executed by GravAD's matched filter function. The GravAD pipeline operates within a predefined frequency range, typically 20 Hz to 1000 Hz, as this is the range where GW signals are predominantly found [28, 29]. The frequency steps within this range are sampled from the GW data signal, ensuring a highly accurate method for frequency analysis.

The same frequency step is employed in constructing a frequency series that is passed to the IMRPhenomD waveform model - through the ripple library. The frequency series, utilized for template generation, starts at 0 Hz and ends at the Nyquist frequency. In the context of our study, the Nyquist frequency was established at 1024 Hz, as our sampling rate was 2048 Hz. In line with the principles of the Fourier Transform, it is imperative that the template, data, and PSD are sampled at the same frequencies, starting from zero. This ensures a consistent frequency step size, which is a prerequisite for accurate data analysis and interpretation [30].

However, the template generation process encounters a challenge. The ripple Python library, utilized for generating templates, demonstrates incompatibility with a frequency series starting at zero. This incompatibility emerges due to the library's unique characteristics for its waveform generation. Consequently, it is necessary to adjust the frequency series to facilitate seamless template generation and avoid NaN values. To tackle this challenge, we avoided searching in the 0 Hz frequency range. To maintain simplicity and prevent significant distortions in the data, we adjusted the frequencies by incrementing them by 1, resulting in a modified frequency series ranging from 1 Hz to 1025 Hz. This seemingly minor shifting has considerable implications for template generation. This shift modifies the frequency content of the templates, which in turn impacts the SNR and the mass values produced by the pipeline. The subtle increase in frequency introduces a slight shift in the template characteristics, thereby influencing the results of the GW data analysis.

Introduction to Automatic Differentiation

AD is a powerful tool for evaluating the derivatives of computational functions, offering significant advantages in numerical optimization tasks. For instance, where the gradient of an objective function needs to be calculated, AD proves invaluable. Primarily, its superiority over numerical differentiation methods lies in its enhanced precision, which effectively mitigates the rounding errors inherent in finite difference approximations [14]. These errors, when accumulated, can significantly impact the overall result. Compared to symbolic differentiation, AD also surpasses its counterpart by avoiding the problem of "expression swell", which can potentially lead to a surge in computations for specific derivatives primarily due to the nature of the product rule [31].

One key feature that sets AD apart is its compatibility with Python, facilitated by the JAX library. AD through JAX negates the need for the explicit programming of the derivative, making AD flexible and easy to implement. In this study, we exploit AD to ascend to the global maximum of the SNR time-series, thus finding the optimal input parameters for template generation. These optimized parameters will characterize the observed GW signal most effectively and thus allow us to extract the signal from the noise during MF. We will perform AD using the grad function from the JAX library, which uses reverse mode AD [18]. This method of AD is suited for many variables' inputs, which our approach takes [32].

The mathematical basis of AD lies in the concept of elementary functions and their derivatives. This approach leverages the fact that every function, regardless of its complexity, can be deconstructed into a sequence of elementary operations, such as addition, multiplication, and trigonometric or exponential functions. The process of AD hinges on the fundamental principles of the chain rule and it ability to take the derivative of elementary functions [33].

Generating a Template

Template generation is primarily accomplished with the Python library ripple [34], which enables waveform template generation using the IMRPhenomD numerical simulations. This model simulates IMR using the phenomenological model consisting of EOB, post-Newtonian (PN) and Numerical-relativity (NR) simulations [35]. The variety of differentiable waveform models available is currently restricted, given the novelty of this technology. Impressively, ripple can produce waveforms in about 0.4 ms on a CPU and, more efficiently, approximately 0.02 ms, on a GPU [15]. This method of waveform generation differs from that used by PyCBC when analyzing the same dataset (GWTC-1), which uses the waveform model SEOBNRv4 opt [26]. This difference could contribute to discrepancies between the computed SNR values from GravAD and PyCBC.

The first step in generating templates is to convert the binary mass values into a singular value known as the chirp mass. The chirp mass Mc is a function of the masses of the two bodies in a binary system (m1 and m2) and is calculated using the formula [36]:

$$M_c = \frac{(m_1 m_2)^{3/5}}{(m_1 + m_2)^{1/5}} \tag{10}$$

The significance of the chirp mass is underscored by its primary role in dictating the frequency evolution, or "chirping", of the GW signal during the inspiral phase of the binary's evolution, caused by the loss of energy due to GWs [37].

Moreover, the symmetric mass ratio, denoted as η, complements the chirp mass in describing the GW signal. It is defined as the ratio of the product of the masses to the square of the total mass. Precisely, η is given by:

$$\eta = \frac{m_1 m_2}{(m_1 + m_2)^2} \tag{11}$$

Ranging between 0 and 0.25, with 0.25 corresponding to equal masses. Different values of η can lead to variations in the IMR phases, influencing the waveform characteristics observed by detectors [35]. The combination of these two parameters allows for the construction of accurate waveform templates that describe merging BHs.

Generating templates is a critical part of the process. Hence, it is of utmost importance that the initial template leads to subsequent meaningful solutions. Initially, both masses for the first template were randomly generated using a pseudo-random number generator from JAX. However, the algorithm demonstrated a propensity for higher mass values, rendering it "top-heavy". To counteract this bias, the initial template was set at the least massive template (22 solar masses), allowing the search to extend into the lighter regions covered by the BHs. The IMRPhenomD waveform model creates a problem with NaN values for waveform templates that have a combined mass below 22 solar masses. These NaN values make it difficult for the grad function to accurately calculate the gradient of the SNR function. Therefore, the only solution is to set a minimum threshold for generating templates.

Using Gradient Descent to Find the Optimal Template

Gradient descent is a widely used optimization technique that can be used with AD to optimize a function. The algorithm works by iteratively updating the mass parameter of the function in the direction of a stationary point. The mass parameters of each binary object are updated at each iteration by taking the product of the learning rate and the gradient and adding a perturbation. The learning rate determines the step size in the update direction, and a lower learning rate can help prevent overshooting the maximum. Our learning rate is randomly generated, selecting a number between 1.5 and 5.5. This choice was made because it allows for exploring a wide search area while maintaining the objective. We avoided using a negative learning rate to avoid regression when converging on a point.

We defined a function to calculate the SNR value of a given template against the data [38]. By using AD, we can take the derivative with respect to the mass parameter and thus obtain the function's gradient. The objective of this function is to influence what the following template generated will be. We use the gradient, the previously discussed learning rate, and a perturbation to update the template waveform parameters. This cycle will repeat as often as instructed to return the optimal template parameters. The template is considered optimal

in the event that it yields the highest SNR value [39].

Waveform template generation through AD and gradient descent offers notable advantages. These methods allow for dynamic generation, eliminating the need for a vast, pre-computed template bank. This process considerably reduces the number of necessary templates for analysis. In addition, templates produced through these techniques can attain a superior fit to the observed data compared to their template bank counterparts, thanks to their tailored optimization for the specific dataset under examination. Despite these advantages, potential drawbacks exist. A significant challenge of our methodology is the tendency to encounter multiple local maxima, complicating the search for the global maximum. This issue can be observed with the template converging at a local rather than the global maximum (Figure 8).
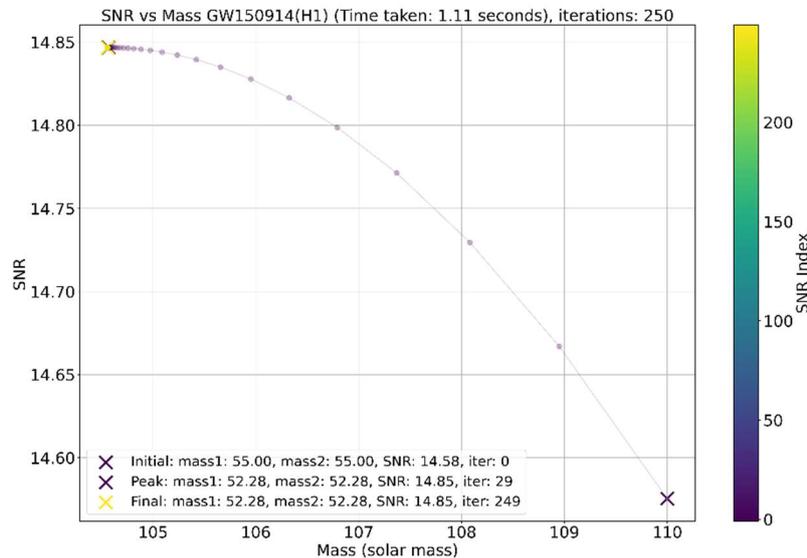

Figure 8. SNR vs Mass graph depicting template convergence at a stationary point.

**Applying Simulated Annealing for Improved Solutions**

To address the challenge of multiple local maxima, our study employs a two-step approach that combines a temperature and annealing rate. We apply simulated annealing, a stochastic optimization technique incorporating random perturbations into the mass computation. GravAD utilizes simulated annealing by adding this perturbation to the product of the gradient and the learning rate. The primary purpose of simulated annealing is to explore the solution space more effectively, enabling the algorithm to discover and traverse multiple stationary points. The algorithm can escape local maxima and potentially find more optimal solutions by introducing random perturbations.

Simulated annealing is inspired by the annealing process in metallurgy, where metals are heated and then gradually cooled to achieve a low-energy state. In our context, the "temperature" parameter controls the degree of randomness introduced into the gradient updates. At high temperatures, the algorithm explores the solution space more aggressively, accepting suboptimal solutions with higher probability. As the temperature decreases, the algorithm becomes more focused on exploiting the current region, converging towards local or global optima. Several tests were done to evaluate the most suitable value when determining the temperature parameter. Higher values work best when there are many local minima, allowing the algorithm to explore more of the search area. On the other hand, a lower temperature value allows for a more focused approach. The annealing rate is another critical factor as it determines the cooling rate of the temperature. The value of the annealing rate is more impactful for lower iteration runs as the number of templates affected by the simulated annealing process is the same regardless of the total iterations [40].

By incorporating simulated annealing into our optimization framework, we aim to enhance the robustness and effectiveness of our methodology. This approach, combined with AD and gradient descent, allows us to generate waveform templates that closely match the observed data, improving our ability to detect and characterize GW sources. The effect on the run time is insignificant as this optimization technique is optimized for JAX operations.

Results

The GravAD pipeline generates data that, when visualized in contour plots (Figure 9), reveals distinct GW signatures. The detection of a GW, illustrated via changing intensity contours, signifies space-time distortions resulting from mergers.

(a). GW170823 present

(b). GW170823 subtracted
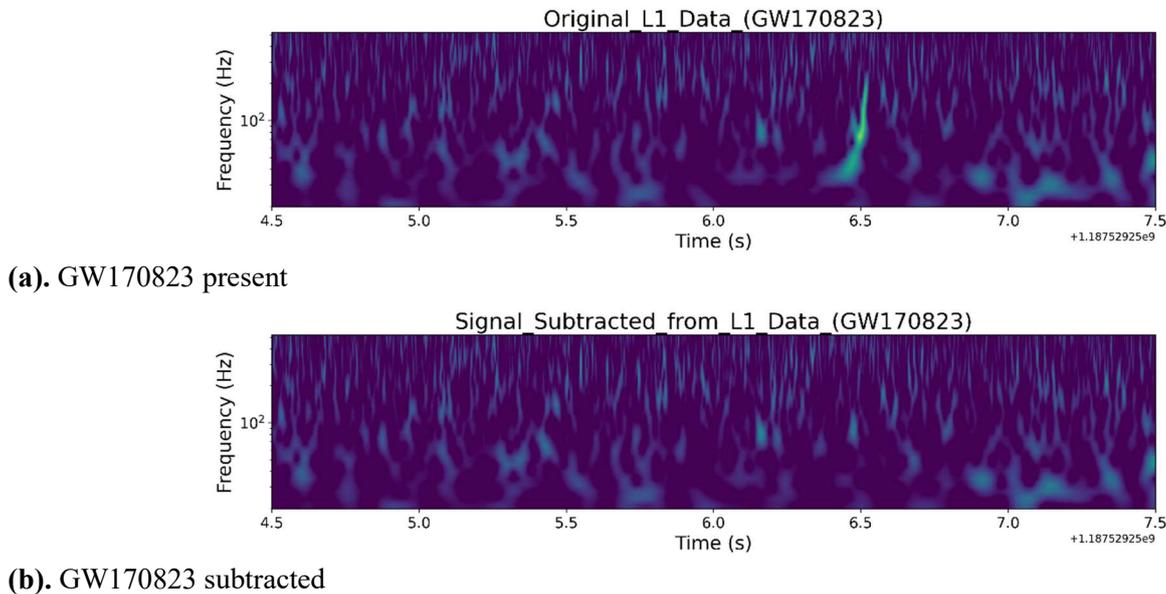
Figure 9. Contour plots of GW170823: (a) with the GW present, illustrating its distinctive signature; (b) with the GW subtracted, highlighting the effectiveness of the GravAD pipeline in isolating these signals.

Comparison to the LIGO Search Pipeline

The Gravitational Wave Transient Catalog (GWTC), a databank concerning GW detections and parameters, employs three pipelines for its GW searches: cWB [41], GstLAL [42], and PyCBC [43]. To evaluate the effectiveness of the AD algorithm against the established methods, we undertook a comparative analysis of the SNR values generated by each search method. The combined SNR was computed as follows:

$$\text{CombinedSNR} = \sqrt{(\text{SNR}_{\text{H1}})^2 + (\text{SNR}_{\text{L1}})^2} \tag{12}$$

The primary objective of this project is to validate and refine the GravAD algorithm by comparing its performance to that of the GWTC, assessing whether the SNR values generated by GravAD align closely to the catalogue. To quantify this alignment, we employ a statistical measure known as the Z-score, which represents the number of standard deviations a data point is from the mean.

The overall Z-score average across all events is -0.78 (Table 1), suggesting a strong degree of similarity between the SNR values produced by GravAD and those obtained from the search pipelines of the GWTC. Nevertheless, it is worth noting that a negative Z-score implies an overall lower SNR value than the mean, indicating that the GravAD method generally yields lower SNR values.

While these initial findings are informative, they mask intricate disparities that emerge when individual events are examined more closely. For example, consider the case of GW151226. The Z-score for this event is -6.61, which denotes a significantly lower SNR value. Delving deeper into the SNR values obtained by each detector for this event (Table 2), it is evident that the GravAD method did not perform as well with the data from the Hanford and Livingston

detectors as it did for other methods.

Conversely, GW170823 demonstrates a Z-score of 2.11, indicating a higher SNR value. Upon examining the individual SNR values for this event, it becomes clear that the GravAD method computed an SNR value of 11.87, outperforming the values of 11.1, 10.8, and 11.5 reported by PyCBC, cWB, and GstLAL, respectively. The observations indicate that the GravAD algorithm, under specific conditions, may provide comparable performance to the established methods utilized by LIGO. However, it is important to note that conventional searches have additional requirements, such as the use of identical templates in both detectors and the application of chi-squared tests [4].

To maintain transparency, the Z-scores were computed using a limited data set provided by the GWTC. This limitation could potentially skew the Z-scoring, leading to potentially biased outcomes. Furthermore, due to the limitations in our Z-score calculation that arise from our implementation of IMRPhenomD and, as a result, restricted search space that the GravAD algorithm can explore, specific GW events were excluded when calculating the average Z-score for a fair evaluation.

Table 1. Z-scores for each GW's SNR, a comparison between the GWTC data and GravAD [44].

Event	PyCBC	cWB	GstLAL	GravAD	Z-Score
GW150914	23.60	25.20	24.40	24.13	-0.33
GW151012	10.00	X	9.50	9.46	-0.81
GW151226	13.10	11.90	13.10	8.12	-6.61
GW170104	13.00	13.00	13.00	13.00	0.00
GW170608	15.40	14.10	14.90	7.77	X
GW170729	9.80	10.20	10.80	9.78	-0.97
GW170809	12.20	X	12.40	12.47	1.18
GW170814	16.00	17.20	15.90	15.94	-0.59
GW170817	30.90	X	33.00	341.88	X
GW170818	X	X	11.30	11.66	-1.00
GW170823	11.10	10.80	11.50	11.87	2.11
AVG					-0.78

Table 2. GWTC SNR values compared to GravAD [44].

Event	PyCBC	cWB	GstLAL	GravAD (H1)	GravAD (L1)	GravAD (combined)
GW150914	23.60	25.20	24.40	19.70	13.94	24.13
GW151012	10.00	X	9.50	6.83	6.54	9.46
GW151226	13.10	11.90	13.10	5.65	5.84	8.12
GW170104	13.00	13.00	13.00	8.79	9.57	13.00
GW170608	15.40	14.1	14.9	5.87	5.09	7.77
GW170729	9.80	10.20	10.80	5.47	8.11	9.78
GW170809	12.20	X	12.40	6.17	10.83	12.47
GW170814	16.00	17.20	15.90	9.05	13.12	15.94
GW170817	30.90	X	33.00	4.95	341.84	341.88
GW170818	X	X	11.30	5.42	10.33	11.66
GW170823	11.1	10.80	11.50	6.85	9.70	11.87

The comparison between the GravAD method and the GWTC pipelines provides valuable insights that can potentially

enhance future GW searches. By indicating a similar range of SNR values for GW events using the same data sets (Figure 10), we can more accurately gauge the effectiveness of the GravAD algorithm. The insights gathered from this comparison will improve our understanding of the algorithm's performance and inform future refinements to enhance its search accuracy.
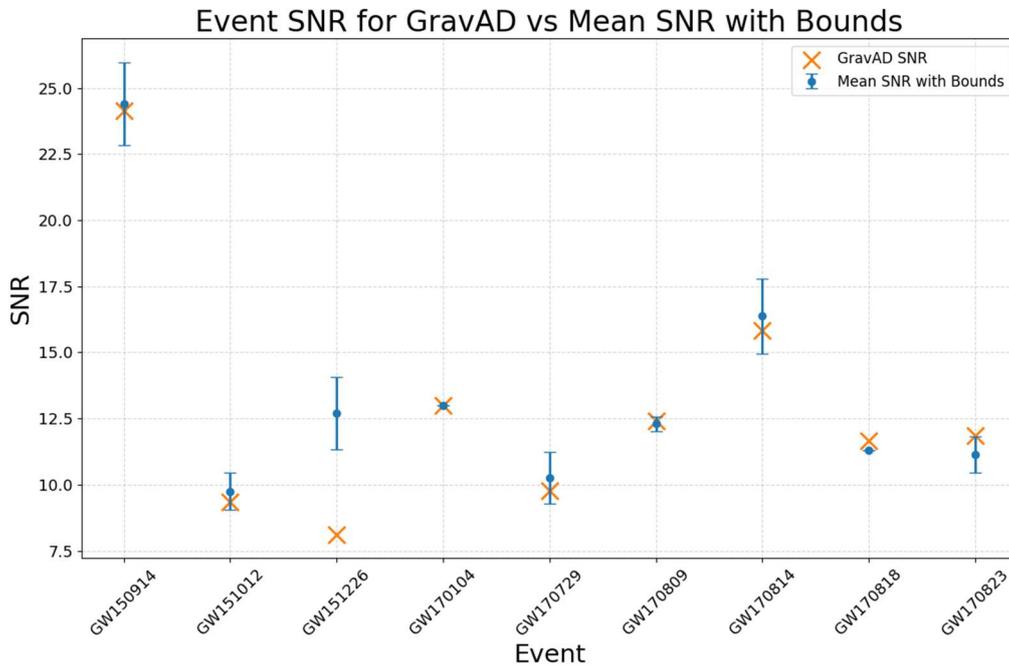

Figure 10. A comparison between GravAD and the mean value for SNR over different events [44].

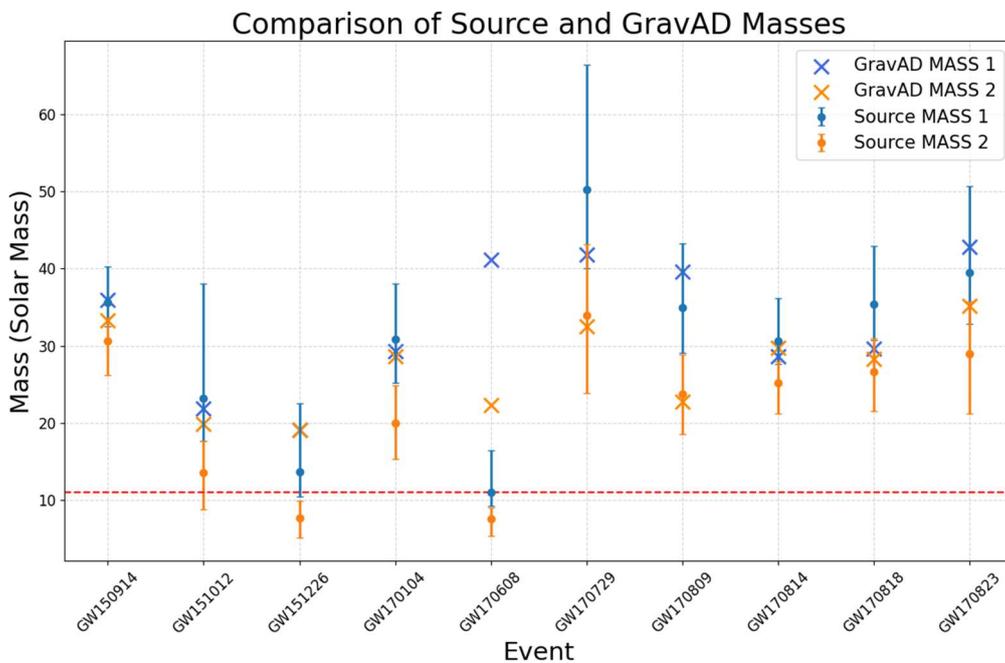**Comparison to the LIGO Parameter Estimation Pipeline**

Although not the primary focus of this project, we consider the mass parameters obtained from the GWTC through the parameter estimation pipelines [44]. This data serves as a benchmark for comparison with the templates generated using AD. To quantitatively assess the correlation between the GWTC results and those produced by GravAD, we utilize the Z-score, a statistical measure. The Z-score average across all events equates to 1.28 (Table 3). This suggests a moderate degree of likeness, indicating that the templates produced by GravAD are reasonably aligned with the mass estimations obtained through the GWTC pipeline. This general observation, however, masks more nuanced disparities at the level of individual events. For instance, consider GW170818, where GravAD predicted a mass of 57.95 solar masses against GWTC's reported value of 59.77 solar masses. This event has a Z-score of -0.54; the mass estimates are remarkably close, differing by a mere 1.82 solar masses. In contrast, the event GW151226 exhibited a Z-score of 3.83, a significantly elevated value which skews the average Z-score of the remainder of the GWs upwards but also indicates room for refining the GravAD algorithm.

Table 3. Mean mass and Z-scores for various GW events - data from GWTC-2 [44].

Event	Mean Mass	GravAD	Z-Score
GW150914	63.23	69.42	1.88
GW151012	37.93	41.78	0.51
GW151226	22.13	38.21	3.83
GW170104	49.27	57.94	1.9
GW170608	18.70	63.58	X
GW170729	81.00	74.29	-0.54
GW170809	56.77	62.38	1.24
GW170814	53.47	58.38	1.75

GW170817	2.80	131.00	X
GW170818	59.77	57.95	-0.42
GW170823	66.30	78.01	1.33
AVG			1.28

Comparing the mass results of GravAD and the parameter estimation masses (Figure 11) reveals that the GravAD algorithm tends to predict higher mass values than those reported by the GWTC. The potential explanations for this tendency could be multifaceted, possibly relating to the unique intrinsic properties of specific GW events, the sensitivity and noise profiles of detectors, and the nuances in the AD methodology employed by the GravAD algorithm. The comparison of GravAD against the parameter estimation pipeline adds to the validity of our search. This is substantiated by the fact that the peak in our SNR search was identified using mass values that closely mirror those of the GWTC. This comparative analysis also paves the way for future investigations through alternative statistical measures and refining the GravAD method to further enhance our results. It is worth noting that certain GW events were excluded from the Z-score calculation due to limitations in our template generation. Despite these exclusions, the insights gained from this study can provide valuable guidance for future research in this domain.

Figure 11. Comparison between source masses (from parameter estimation) and GravAD's predicted masses. The red line illustrates the minimum mass threshold from which GravAD cannot search below.

Finding the Optimal Template

A significant part of this research involves identifying the optimal template by altering the mass parameter in a coordinated search using the GravAD algorithm and comparing this with the main LIGO search pipelines: PyCBC, GstLAL, and cWB. A key measure of success in this study is the ability to reduce the number of templates used in the search compared to the conventional LIGO search pipelines. Although no comprehensive compilation of data that quantifies the exact number of templates utilized for each GW event and pipeline exists, an estimate suggests that approximately $N \sim 5 \times 10^5$ templates are employed [45]. Nonetheless, we have managed to gather data on utilizing GravAD templates, presented in Table 4.

The data in Table 4 offers several noteworthy insights. Firstly, the case of GW151226 stands out due to the stark contrast in the number of iterations required between the two detectors. Using the data from the Hanford detector, GravAD predicts the highest SNR value at the initial template, whereas the Livingston data reaches this peak only on the 365th iteration. Despite this considerable discrepancy, both detectors report a similar SNR value. This incongruity points towards the need for further refinement in GravAD's methodology. The nature of GW151226,

with its mean mass being 22.13, may account for this disparity. It is just about replicable by GravAD's implementation of IMRPhenomD, suggesting that GravAD may not be as well suited to low-mass black holes as it is for higher-mass black holes. This observation reinforces the earlier discussion regarding the limitations of GravAD in accurately assessing certain types of GW events.

Table 4. Number of iterations GravAD required to find peak SNR value.

| Event	Strain	Iteration
GW150914	H1	37
GW150914	L1	72
GW151012	H1	9
GW151012	L1	64
GW151226	H1	0
GW151226	L1	365
GW170104	H1	110
GW170104	L1	20
GW170608	H1	122
GW170608	L1	496
GW170729	H1	42
GW1700729	L1	499
GW170809	H1	449
GW170809	L1	428
GW170814	H1	29
GW170814	L1	19
GW170817	H1	0
GW170817	L1	492
GW170818	H1	196
GW170818	L1	210
GW170823	H1	132
GW170823	L1	203
AVG		181.55

Secondly, the data for GW150914 shows a relatively small disparity in the number of templates used between detectors, with 37 and 72 iterations, respectively. This data suggests that it would be superfluous to continue searching and generating templates up to 500, as both peaks have already been located. Such information could be utilized to fine-tune GravAD by devising a system that dynamically adjusts the total number of templates produced based on the direction the algorithm is progressing. For instance, at the 37th iteration, GravAD detected the peak. After an additional 50 iterations, it had yet to find a better value. This could prompt the introduction of a mechanism to halt the generation process, or restart the search from the peak position, and examine how resetting the simulated annealing temperature might influence the search trajectory.

Lastly, Table 4 suggests that data from the Hanford detector typically locates the peak SNR with fewer template generations than the Livingston detector. As these interferometers continue to improve over time and the data signal becomes clearer, fewer template iterations will likely be required, thereby enhancing the efficiency of the GravAD algorithm. This finding provides an optimistic outlook for the future of GW detection and the role of the GravAD algorithm in this evolving field.

Assessing the Robustness of the Approach

For GravAD, we can define its robustness by the algorithm's ability to deliver accurate and reliable results across a wide range of conditions, including varying noise levels (from different detectors), different types of gravitational waveforms, and different numbers of iterations. The robustness of our methodology is apparent across a spectrum of SNRs and mass parameters. In certain instances, as the iteration count escalated, the precision of the optimal template correspondingly improved, suggesting a robust nature of the algorithm. To further scrutinize the resilience of our approach, it would be beneficial to examine more GWs and data from other detectors than Hanford and Livingston. This would allow GravAD's performance across a diverse range of GWs and scenarios to be tested.

Notably, our algorithm exhibits a discernible convergence towards a certain mass (Figure 12). This predictable trajectory suggests that the algorithm maintains its stability and accuracy even as iteration counts increase, indicating a high degree of robustness. However, it should be noted that this level of convergence is typically observed in optimal conditions. In a more commonplace scenario, the GravAD pipeline is likely to converge towards a secondary maximum rather than the global maximum. The underlying cause of this observed behavior remains a topic of investigation, with potential contributing factors being the simulated annealing algorithm or the gradient itself. Despite these challenges, the pipeline continues to demonstrate its capability to identify the global maximum (Figure 13). This resilience in the face of potential suboptimal conditions further underscores the robustness of our approach.
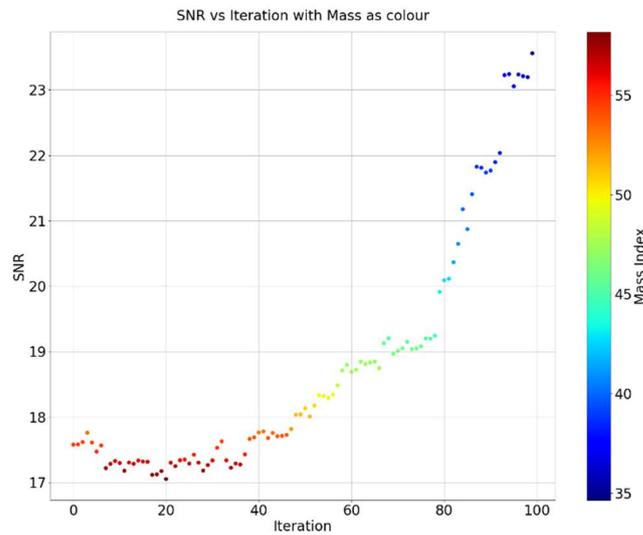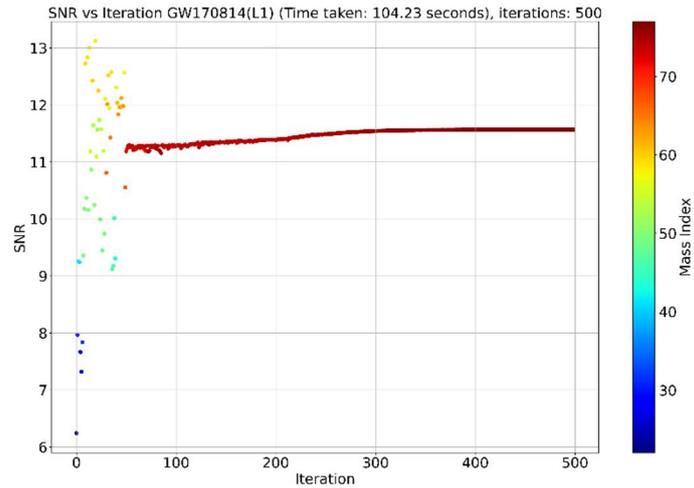
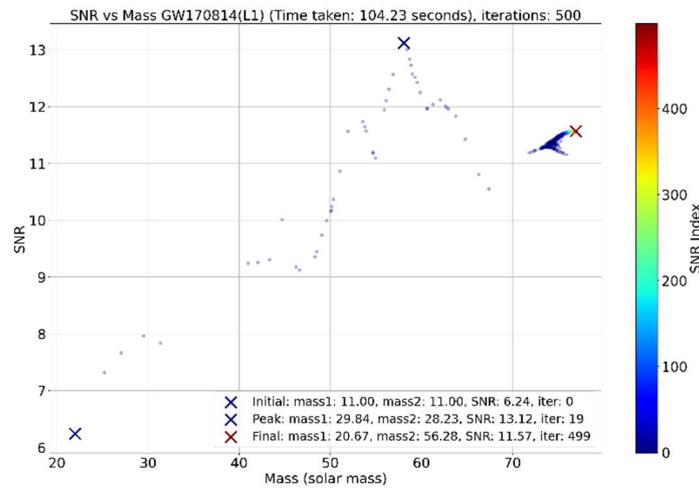

(a). 100 Template GravAD run
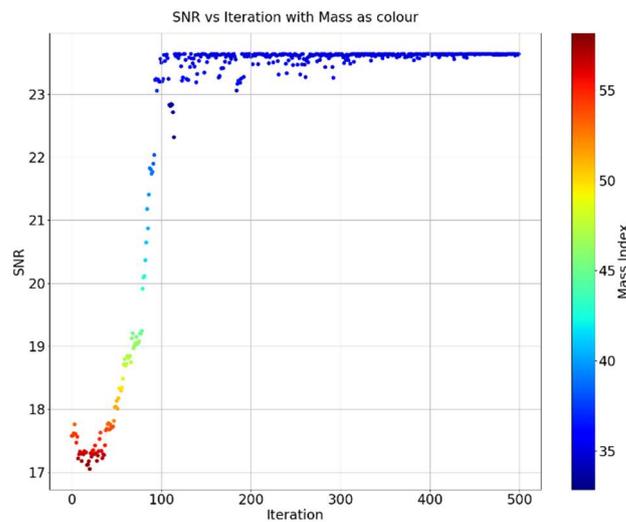
(b). 500 Template GravAD run

Figure 12. GravAD's convergence with higher iterations. The graphs demonstrate SNR vs iteration. As the number of templates is increased, a convergence to the peak SNR value appears.

(a). SNR vs Iteration

(a). SNR vs Mass

Figure 13. GravAD's tendency to converge at secondary local maxima.

One of the central elements contributing to this robustness is the pipeline's ability to iterate and refine its results. As the number of iterations grow, the likelihood of identifying and converging on a global optimum increases, suggesting an innate resilience in the algorithm's design. This ability to persist and refine through iterations is a powerful tool in searching for GWs, contributing to the overall robustness of the approach.

Upon evaluating various initial conditions for our algorithm, we can gauge the effectiveness of GravAD by determining its consistency in converging to the same SNR value, as depicted in Figure 14. Although certain outliers are present, a broader view indicates that GravAD is capable of identifying the optimal template irrespective of the initial point. This invariably leads to a peak in the SNR, further establishing the algorithm's efficiency and robustness.
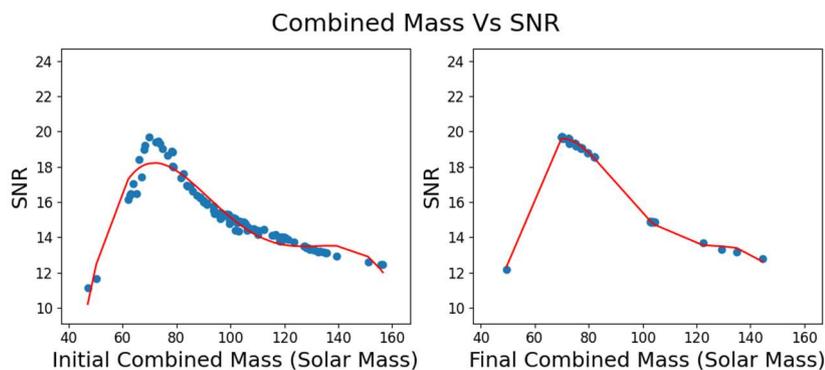

Figure 14. A comparison between the initial and optimal templates (using 100 starting points and end points). The graph on the left shows the initial template used by GravAD, while the graph on the right displays the optimal template obtained from these initial starting values. The red line represents the fitted regression line.

Effectiveness of JIT Optimization

The integration of JIT compilation in our computational processes has substantially reduced run-time, thereby speeding up both the generation and analysis of waveform templates. An interesting characteristic is observed in the first iteration of template generation and analysis, which is more time-intensive than the subsequent iterations. This observation demonstrates the effectiveness of the JIT optimization offered by the *jax.jit* function.

When contrasting the time discrepancy between test cases of JIT-optimized GravAD and its non-optimized counterpart (Figures 15), we notice a significant increment in time for the latter. This disparity in computation time between the two scenarios clearly underlines the efficiency of JIT optimization.
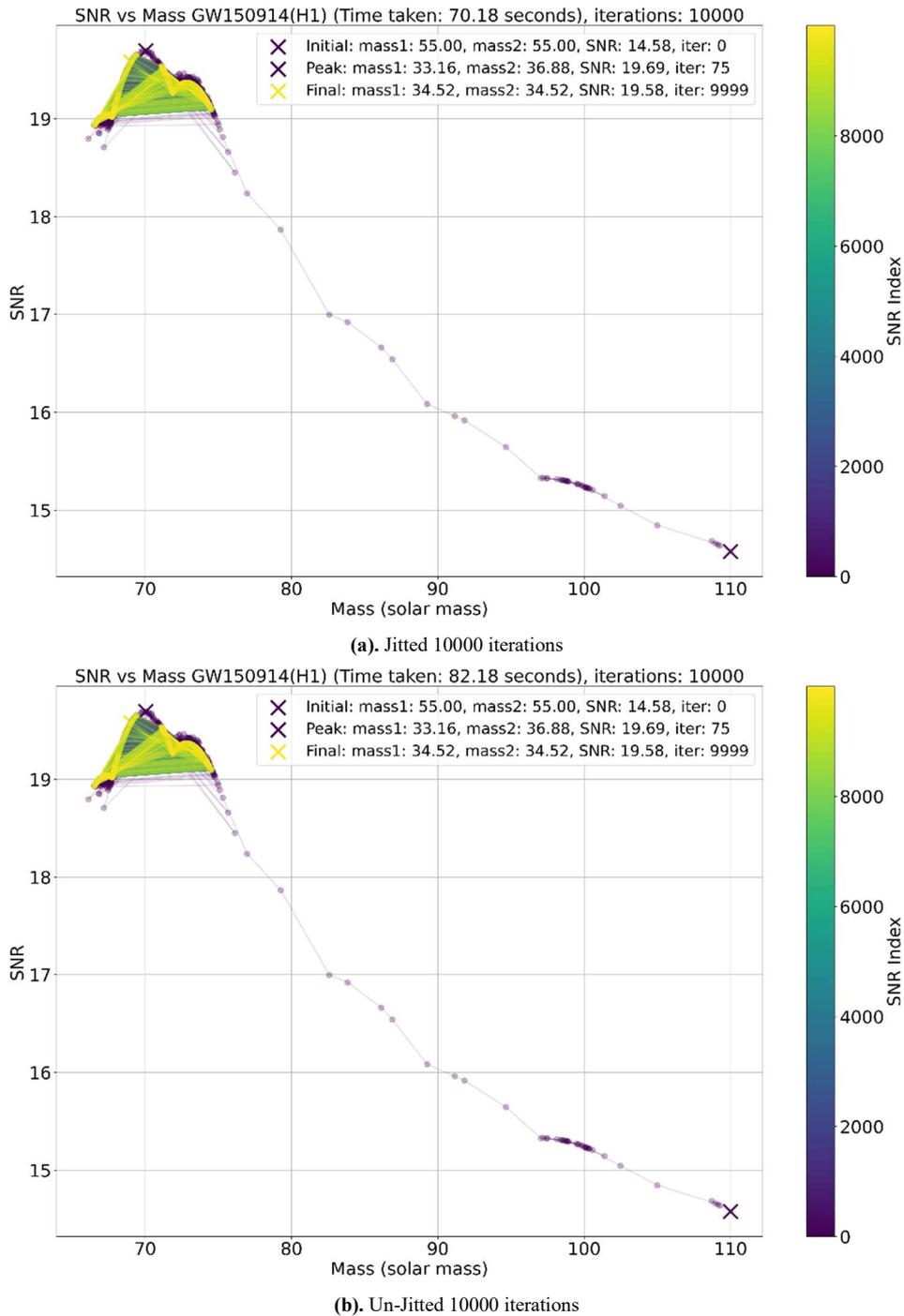

(a). Jitted 10000 iterations

(b). Un-Jitted 10000 iterations

Figure 15. SNR vs Mass with color as the iterations demonstrating the effectiveness of JIT compilation. Generated using a T4 GPU.

Leveraging Hardware Configurations with JAX

GravAD is written in entirely JAX compatible code. This allows for GPU hardware acceleration, a major triumph in analyzing GW signals. While the AD method yields similar SNRs and mass outcomes to those of the LIGO pipelines, it accomplishes these results with remarkable efficiency. For instance, the AD method, using merely 500 templates, can generate results for GW150914 in approximately 0.9 seconds, in stark contrast to the 250,000 templates used by the LIGO pipelines [4]. This impressive computational efficiency positions the AD method as a valuable tool for preliminary testing and other rapid-result contexts. The dynamic template generation capability intrinsic to the AD method bolsters this efficiency allowing for 500 templates to be generated dynamically in response to the incoming signal. This streamlined approach contributes significantly to the reduction in computational load. However, direct time comparisons with other pipelines present challenges due to hardware variations and differing operational conditions.

The remarkable potential of the AD method is further amplified when GravAD is run on GPUs using JAX, leading to a performance enhancement (Figure 16). Utilizing powerful hardware such as an Nvidia A100 through Google Colab enables GravAD to calculate the SNRs of various GWs in less than a second. This represents a substantial improvement in computational efficiency over CPU hardware such as the Intel Xeon CPU. This ability to leverage powerful computing systems demonstrates the scalability and versatility of GravAD. It reinforces the potential of the AD method to contribute significantly to the field by providing fast, accurate results, even when the number of templates is reduced, thereby enhancing the overall computational efficiency.

Overall, the ability to dynamically generate templates in response to incoming signals and leverage different hardware configurations significantly reduces the computational load of GravAD.
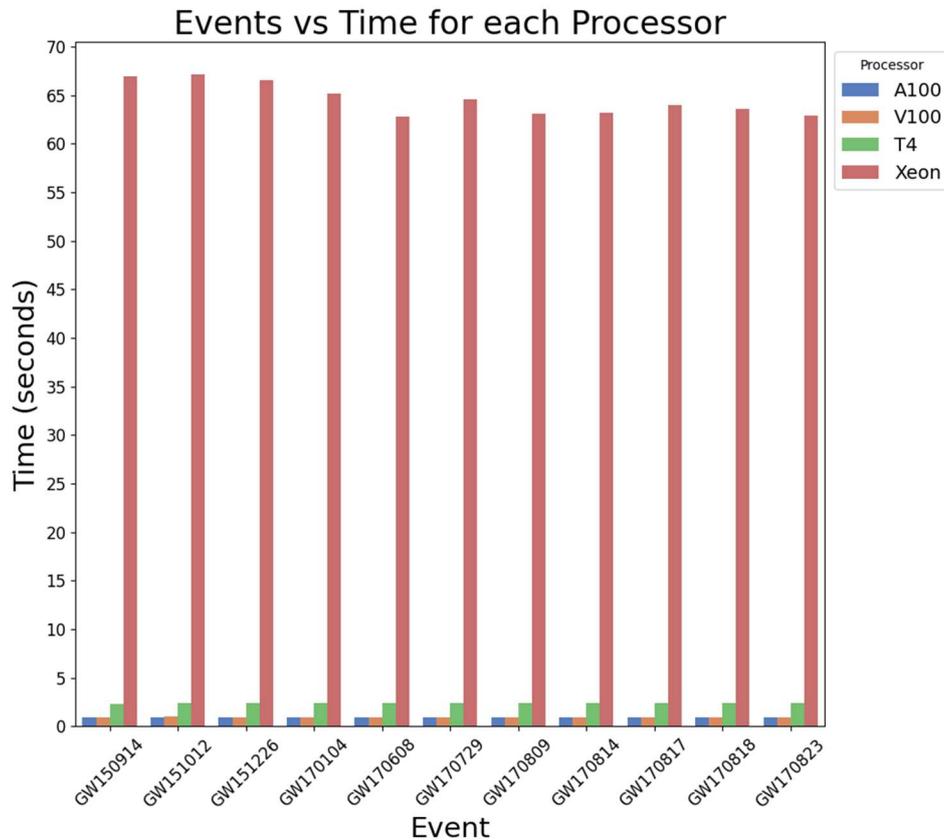
Figure 16. Run-time for a 500 iteration GravAD run, which includes tasks such as template generation, matched filtering, and SNR computation. The experiment was performed on various hardware configurations, including A100 (GPU), V100 (GPU), T4 (GPU), and Intel Xeon (CPU).

Discussion

The GravAD algorithm has demonstrated its potential in GW detection through the close alignment of its Z-scores for SNR and mass values compared to established methods. While the trade-off between SNR potency and evaluation speed may initially appear as a limitation, this characteristic makes GravAD well-suited for specific applications

within the GW research field. The ability of GravAD to rapidly evaluate potential GW events allows it to serve as an efficient preliminary tool. Given the vast amount of data collected by GW detectors, having a method that can quickly assess whether a given signal is worth further analysis can significantly streamline the detection process. By identifying promising candidates for subsequent, more computationally intensive analysis, GravAD enables researchers to allocate resources more effectively. Ultimately, GravAD could be used as a first-pass evaluation before further analysis by other pipelines.

One major limitation of this study is the reliance on simulated annealing as an optimization method, which does not always converge to the peak SNR value. Although simulated annealing has proven to be an effective optimization technique, its performance can be sensitive to the choice of temperature schedule and initial conditions. Additionally, it may be susceptible to getting stuck in local optima or taking longer than necessary to converge. To address this limitation, future research could investigate implementing alternative optimization algorithms, such as momentum-based algorithms, which could outperform simulated annealing in specific applications. Momentum-based algorithms, such as Adaptive Moment Estimation (Adam) [46], could help prevent the optimizer from getting stuck in local optima and accelerate convergence. Additionally, combining multiple optimization algorithms could enhance the overall performance and reliability of the search process. As our results have indicated, GravAD can generate an optimal template on average in $N \sim 180$ iterations. This highlights the potential for improving the algorithm's efficiency through further code optimization. By employing advanced programming techniques and methodologies, it may be possible to reduce further the number of iterations required, thereby decreasing the computational time and making the algorithm even more efficient.

Another area for improvement of this study is its inability to handle binary NS and low-mass BH systems due to the minimum mass constraint of the IMRPhenomD waveform model used by the ripple software. This constraint renders the current methodology ineffective for searching for GWs from these mergers, which are both critical astrophysical sources of GWs. Future research should explore alternative waveform models that can handle a broader range of mass ratios and compact object types to address this limitation.

One of our work's most significant potential extensions is the adaptation of GravAD for real-time data analysis. Achieving this would enable GravAD to contribute directly to detecting and analyzing new GW events as they occur. GravAD could enable the scientific community to respond more quickly and efficiently to new GW detections by providing near real-time evaluations and facilitating rapid follow-up observations and research.

1. Conclusion

In this research, we pursued two main objectives: the development of GravAD, a custom search pipeline written in Python and compatible with JAX, as a novel approach to gravitational wave detection and its benchmarking against existing search pipelines. Through our investigation, we have made significant advancements in gravitational wave detection and addressed key limitations of established methodologies. Our results demonstrate that GravAD enables the analysis of gravitational wave signals using significantly fewer templates compared to conventional methods. This reduction in template analysis not only improves computational efficiency but also maintains a comparable level of performance, thus streamlining the detection process.

It is essential to acknowledge the limitation of GravAD's minimum mass constraint, which restricts its application in the search for GW signals from neutron stars and low-mass black holes. To overcome this limitation, future studies should focus on incorporating waveform models encompassing a more comprehensive range of mass ratios and compact object types, allowing for a more comprehensive analysis.

Moving forward, our research directions will concentrate on further refining GravAD to enhance its robustness and efficiency. By doing so, we aim to enable rapid computations and free up valuable resources for other scientific pursuits. One of the major advantages of GravAD is its low computational cost, which enables the swift analysis of data. This feature not only enhances the efficiency of GW analysis but also allows researchers to allocate more resources to other areas of study.

In summary, our research has resulted in the development of GravAD, a promising method that offers a comparable level of performance to existing search pipelines while significantly reducing the number of templates required for analysis. By overcoming limitations and pursuing future refinements, GravAD has the potential to revolutionize GW detection and contribute to a deeper understanding of the Universe's gravitational wave signals.

References

[1] Cutler, Curt, and Kip S. Thorne. "An overview of gravitational-wave sources." *General Relativity and Gravitation* (2002): 72-111.

[2] Mitchell, J. (1784) 'Vii. on the means of discovering the distance, magnitude, &c. of the fixed stars, in consequence of the diminution of the velocity of their light, in case such a diminution should be found to take place in any of them, and such other data should be procured from observations, as would be farther necessary for that purpose. by the rev. John Michell, B.D. F.R.S.. in a letter to Henry Cavendish, Esq.. F.R.S. and A.S', *Philosophical Transactions of the Royal Society of London*, 74, pp. 35–57. doi:10.1098/rstl.1784.0008.

[3] A. Einstein. "Die Grundlage der allgemeinen Relativitätstheorie". In: Annalen der Physik 354.7 (1916), pp. 769–822.

[4] B. P. Abbott et al. "Observation of Gravitational Waves from a Binary Black Hole Merger". In: Phys. Rev. Lett. 116 (6 Feb. 2016), p. 061102

[5] Stephen W. Hawking and Werner Israel. Three Hundred Years of Gravitation. Provided by the SAO/NASA Astrophysics Data System. Cambridge University Press, 1989, pp. 330–458.

[6] B. P. Abbott et al. "GW150914: The Advanced LIGO Detectors in the Era of First Discoveries". In: Phys. Rev. Lett. 116 (13 Mar. 2016), p. 131103.

[7] Rainer Weiss. "Republication of: Electromagnetically coupled broadband gravitational antenna". In: General Relativity and Gravitation 54.11 (2022)

[8] A. Abramovici et al. "LIGO: The Laser Interferometer Gravitational-Wave Observatory". eng. In: Science (New York, N.Y.) 256.5055 (Apr. 1992), pp. 325–333

[9] Maite Mateu-Lucena et al. "Parameter estimation with the current generation of phenomenological waveform models applied to the black hole mergers of GWTC-1". In: Monthly Notices of the Royal Astronomical Society 517.2 (Oct. 2022), pp. 2403–2425

[10] Andrea Taracchini et al. "Effective-one-body model for black-hole binaries with generic mass ratios and spins". In: Phys. Rev. D 89 (6 Mar. 2014), p. 061502

[11] Bruce Allen et al. "FINDCHIRP: An algorithm for detection of gravitational waves from inspiraling compact binaries". In: Physical Review D 85.12 (June 2012)

[12] Alex Nitz et al. Gwastro/pycbc-tutorials: Learn how to use PYCBC to analyze gravitational- wave data and do parameter inference.

[13] Samantha A Usman et al. "The PyCBC search for gravitational waves from compact binary coalescence". In: Classical and Quantum Gravity 33.21 (Oct. 2016), p. 215004.

[14] Adam Coogan et al. "Efficient gravitational wave template bank generation with differentiable waveforms". In: Phys. Rev. D 106 (12 Dec. 2022), p. 122001

[15] Thomas D. P. Edwards et al. ripple: Differentiable and Hardware-Accelerated Waveforms for Gravitational Wave Data Analysis. 2023

[16] B P Abbott et al. "A guide to LIGO–Virgo detector noise and extraction of tran- sient gravitational-wave signals". In: Classical and Quantum Gravity 37.5 (Feb. 2020), p. 055002

[17] Michele Maggiore et al. "Science case for the Einstein telescope". In: Journal of Cos- mology and Astroparticle Physics 2020.03 (Mar. 2020), p. 050

[18] James Bradbury et al. JAX: composable transformations of Python+NumPy programs. Version 0.3.13. 2018

[19] Roberto Vio and Paola Andreani. Everything you always wanted to know about matched filters (but were afraid to ask). 2021.

[20] Vinh Nguyen. "Gravitational Wave Detection: Event GW170814 and the performance of matched filtering and chi-squared consistency testing in the presence of Gaussian noise, non-Gaussianly distributed noise, and transient glitches". In: Pioneer Research Journal (2020)

[21] Benjamin J. Owen and B. S. Sathyaprakash. "Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement". In: Phys. Rev. D 60 (2 June 1999), p. 022002.

[22] S Babak et al. "A template bank to search for gravitational waves from inspiralling compact binaries: I. Physical models". In: Classical and Quantum Gravity 23.18 (Aug. 2006), pp. 5477–5504.

[23] E Oran Brigham. The fast Fourier transform and its applications. Prentice-Hall, Inc., 1988.

[24] Duncan A. Brown. "Searching for Gravitational Radiation from Binary Black Hole MACHOs in the Galactic Halo". In: (2007).

[25] Alexander H. Nitz et al. "Detecting Binary Compact-object Mergers with Gravitational Waves: Understanding and Improving the Sensitivity of the PyCBC Search". In: The Astrophysical Journal 849.2 (Nov. 2017), p. 118.

[26] B. P. Abbott et al. "GWTC-1: A Gravitational-Wave Transient Catalog of Compact Binary Mergers Observed by LIGO and Virgo during the First and Second Observing Runs". In: Physical Review X 9.3 (Sept. 2019).

[27] Alex Sanchez-Stern et al. Finding Root Causes of Floating Point Error with Herbgrind. 2018

[28] R. Abbott et al. "GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo during the First Half of the Third Observing Run". In: Physical Review X 11.2 (June 2021).

[29] Michele Maggiore. Gravitational Waves: Volume 1: Theory and Experiments 1. 1st ed. Oxford University Press, 2008. isbn: 9780198570745; 0198570740.

[30] David G. Blair. The Detection of Gravitational Waves. 2005, pp. 406–451. isbn: 0521021022; 9780521021029.

[31] Atilim Gunes Baydin et al. Automatic differentiation in machine learning: a survey. 2018

[32] Charles C. Margossian. "A review of automatic differentiation and its efficient imple- mentation". In: WIREs Data Mining and Knowledge Discovery 9.4 (Mar. 2019).

[33] Richard D. Neidinger. "Introduction to Automatic Differentiation and MATLAB Object- Oriented Programming". In: SIAM Review 52.3 (2010), pp. 545–563.

[34] Ted Edwards et al. Ripple. GitHub repository. Accessed May 3, 2023.

[35] Sebastian Khan et al. "Frequency-domain gravitational waves from nonprecessing black-hole binaries. II. A phenomenological model for the advanced detector era". In: Phys. Rev. D 93 (4 Feb. 2016), p. 044007.

[36] Curt Cutler and Éanna E. Flanagan. "Gravitational waves from merging compact binaries: How accurately can one extract the binary's parameters from the inspiral waveform?" In: Physical Review D 49.6 (Mar. 1994), pp. 2658–2697.

[37] Bernard Schutz. A First Course in General Relativity. 2nd. Cambridge University Press, June 2009. isbn: 0521887054.

[38] BINARY BLACK HOLE SIGNALS IN LIGO OPEN DATA. Sept. 2017. url: https: //gwosc.org/s/events/GW150914/GW150914_tutorial.py.

[39] Sebastian Ruder. An overview of gradient descent optimization algorithms. 2017.

[40] A.G. Nikolaev and Sheldon Jacobson. "Simulated Annealing". In: vol. 146. Sept. 2010, pp. 1–39. isbn: 978-1-4419-1663-1

[41] S. Klimenko et al. "Method for detection and reconstruction of gravitational wave transients with networks of advanced detectors". In: Phys. Rev. D 93 (4 Feb. 2016), p. 042004.

[42] Kipp Cannon et al. GstLAL: A software framework for gravitational wave discovery. 2020.

[43] Alex Nitz et al. gwastro/pycbc: v2.1.2 release of PyCBC. Version v2.1.2. May 2023.

[44] Rich Abbott et al. "Open data from the first and second observing runs of Advanced LIGO and Advanced Virgo". In: SoftwareX 13 (Jan. 2021), p. 100658.

[45] Michael Pürrer. "Frequency domain reduced order model of aligned-spin effective-one- body waveforms with generic mass ratios and spins". In: Phys. Rev. D 93 (6 Mar. 2016), p. 064041.

[46] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. 2017.

2. Appendix

For the full GravAD code visit: https://github.com/WDoyle123/GravAD