The Pennsylvania State University
The J. Jeffrey and Ann Marie Fox Graduate School

ADVANCEMENTS IN AND OPERATIONS OF GRAVITATIONAL-WAVE SEARCHES DURING THE FOURTH OBSERVING RUN

A Dissertation in
Physics
by
Prathamesh Joshi

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2025

The dissertation of Prathamesh Joshi was reviewed and approved by the following:

Chad Hanna
Professor of Physics & Astronomy and Astrophysics
Dissertation Advisor
Chair of Committee

Bangalore Sathyaprakash
Elsbach Professor of Physics & Professor of Astronomy and Astrophysics

Donghui Jeong
Professor of Astronomy and Astrophysics

Sarah Shandera
Professor of Physics

Irina Mocioiu
Associate Professor of Physics
Director of Graduate Studies

Abstract

Gravitational Waves were first predicted by Einstein more than a hundred years ago. Often described as ripples in the fabric of spacetime, they arose as wave solutions to his field equations in the theory of general relativity. The existence as well as detectability of these gravitational waves was long in doubt, most notably by Einstein himself. The question of existence was answered in 1974 with the detection of the Hulse–Taylor binary, two neutron stars revolving around each other and losing energy precisely at the rate predicted by gravitational waves.

Since this was an indirect observation of gravitational waves, the question of detectability still remained. It was finally answered in 2015 when the LIGO-Virgo-KAGRA collaboration directly detected gravitational waves from a binary black hole system, since named GW150914. This detection was significant, since it kick-started a completely new field of physics, gravitational wave astronomy. We could now observe the universe in a completely new way, using gravitational waves instead of the traditional electromagnetic means.

Since then, around 250 new gravitational wave candidates have been detected. The question of gravitational wave detection has been conclusively answered. This does not mean that the field of gravitational wave astronomy is over. On the contrary, as we detect more gravitational waves, more questions arise. These include questions about black hole and neutron star formation mechanisms and properties, merger rates and population statistics of these astronomical objects, and the implications of these on general relativity and cosmology.

Gravitational wave searches are the first step of extracting scientific results from gravitational wave data. Low-latency searches can contribute to multi-messenger detections, greatly increasing the amount of information we can extract from astronomical events, whereas high-latency searches can be used to aggregate results from multiple robust detections. Hence, in order to answer the previous questions, it is imperative to make gravitational wave searches more sensitive, robust, accurate, and computationally efficient.

This work describes the results of such efforts in the lead up to, and during the fourth observing run of the LIGO Scientific, Virgo and KAGRA Collaboration, with a particular emphasis on the GstLAL search pipeline. It also talks about both the low-latency and high-latency operations of the GstLAL pipeline which have contributed towards official results of GstLAL as well as the LVK collaboration for the fourth observing run.

Table of Contents

List of Figures ix

List of Tables xix

Acknowledgments xxi

Chapter 1
Introduction 1

1.1 Gravitational Wave Formalism in General Relativity . . . 1
1.1.1 Linearized Field Equations and Wave Solutions . . . 2
1.1.2 Transverse-Traceless Gauge and Polarization Modes . . . 3
1.1.3 Amplitude of Gravitational Waves . . . 4
1.2 Sources of Gravitational Waves . . . 4
1.2.1 Compact Binary Coalescences (CBCs) . . . 4
1.2.2 Core-Collapse Supernovae and Other Transient Sources . . . 5
1.2.3 Continuous Gravitational Wave Sources . . . 5
1.2.4 Stochastic Background of Gravitational Waves . . . 6
1.3 Gravitational Wave Detectors . . . 6
1.3.1 Interferometer-Based Detectors . . . 6
1.3.2 Noise Sources and Mitigation . . . 7
1.3.3 Future Developments . . . 8
1.3.4 Data Analysis and Signal Extraction . . . 8

Chapter 2
How Many Times Should We Matched Filter Gravitational Wave Data? A Comparison of GstLAL's Online and Offline Performance 10

2.1 Introduction . . . 11
2.2 Software . . . 12
2.2.1 General GstLAL methods . . . 12
2.2.2 Online GstLAL Analysis . . . 14
2.2.3 Offline GstLAL Analysis . . . 16
2.3 Methodology . . . 17
2.3.1 Online Rank . . . 17

2.3.2 Offline Rank Stage Methods 18
2.3.3 Dropped Data Refiltering 20
2.3.4 Computational Cost Reduction 21
2.4 Results 22
2.4.1 MDC Data Set and Analyses 22
2.4.2 Sensitivity Comparisons 22
2.4.3 Candidate Lists 24
2.4.4 Injection Parameter Recovery Comparisons 25
2.5 Conclusion 27
2.6 Acknowledgements 29

Chapter 3
GstLAL's online operations during the fourth observing run 31
3.1 O4a 32
3.1.1 Public Alerts 32
3.1.2 GW230529 33
3.2 O4b 33
3.2.1 AllSky 34
3.2.1.1 HL vs HLV 34
3.2.1.2 Results 35
3.2.2 Early Warning 36
3.2.2.1 Data whitening problem 36
3.2.2.2 Results 37
3.2.3 SSM 37
3.2.3.1 Results 37
3.2.4 Esme 37
3.2.4.1 Results 39
3.3 Conclusion 39

Chapter 4
New Methods for Offline GstLAL Analyses 41
4.1 Introduction 41
4.2 Overview of the GstLAL Offline Analysis 42
4.2.1 Template bank creation stage 43
4.2.1.1 AllSky template bank 43
4.2.1.2 IMBH template bank 44
4.2.1.3 Population model 44
4.2.2 power spectral density (PSD) measurement stage 45
4.2.3 Singular value decomposition (SVD) of templates and template whitening stage 46
4.2.4 Matched filtering stage 46
4.2.5 Rank stage 47
4.3 New Methods 48
4.3.1 Online Rank 48

4.3.2 Dropped Data Refiltering 48
4.3.3 intermediate-mass black hole (IMBH) Analysis 49
4.3.4 Combining the AllSky and IMBH analyses 51
4.3.5 Modularity and reusability of results 53
4.3.6 Ranking statistic improvements 53
4.3.7 New Extinction Model 55
4.4 Results 58
4.4.1 Data set 58
4.4.2 Candidate lists 58
4.4.3 Sensitivity comparison 58
4.4.4 Performance of individual features 61
4.4.4.1 IMBH analysis 61
4.4.4.2 Extrapolation in $\rho - \xi^2$ noise model 61
4.4.4.3 Reliability of results 62
4.5 Conclusion 62

Chapter 5
Method for removing signal contamination during significance estimation of a GstLAL anaysis 70
5.1 Introduction 70
5.2 Signal Contamination 71
5.2.1 Likelihood Ratio 71
5.2.2 The $\rho - \xi^2$ histograms 72
5.3 Removing contamination with the Background Filter 73
5.3.1 Recording events 75
5.3.2 Removing contamination 76
5.4 Results 78
5.4.1 Analysis methods 78
5.4.2 Simulation Set 79
5.4.3 Sensitivity Improvements 81
5.5 Conclusion 82
5.6 Acknowledgements 84
5.7 Appendix A: Choice of constraints, and their impact on performance . . 85
5.8 Appendix B: Criteria for removing events from the background, and its effect on sensitivity 85
5.9 Appendix C: Differing impacts of singal contamination of the sensitivities of template bins 88

Chapter 6
Metric Assisted Stochastic Sampling (MASS) search for gravitational waves from binary black hole mergers 92
6.1 Introduction 92
6.2 Motivation 93
6.3 Methods 94

6.3.1 Data 95
6.3.2 Spectrum estimation and whitening 96
6.3.3 Simulation capabilities 97
6.3.4 Parameter space sampling 97
6.3.4.1 Computation of the binary parameter space metric . . . 98
6.3.4.2 Choice of coordinates 100
6.3.4.3 Pathologies of the numerical metric 100
6.3.4.4 Drawing random samples from $\Theta(\delta_k, \vec{\lambda})$ 101
6.3.4.5 Parameter space constraints 102
6.3.4.6 Glitch Rejection 102
6.3.4.7 Computing the log-likelihood ratio, $\mathcal{L}$ 103
6.3.5 Background estimation 106
6.4 Results 107
6.4.1 Data set 107
6.4.2 Search parameter space 107
6.4.3 Simulation set 107
6.4.4 Candidate list 107
6.4.5 Sensitivity estimate 111
6.5 Conclusion 113
6.6 Acknowledgements 114
6.7 Appendix A: Data release details and code versions 114
6.8 Appendix B: Follow-up of missed injections 114

Chapter 7
Method to get Better Sky Maps in a GstLAL Low-Latency Analysis 116
7.1 Introduction 117
7.2 Methodology 120
7.2.1 General GstLAL Methods 120
7.2.2 SNR Optimizer Methods 121
7.2.2.1 Design Principles 121
7.2.2.2 Template Bank 122
7.2.2.3 Search Algorithm 122
7.2.2.4 Coincidence Formation 125
7.2.2.5 O4 Configuration 127
7.3 Results 129
7.3.1 Data set 129
7.3.2 SNR improvement 130
7.3.3 Sky map improvement 130
7.3.4 Latency 134
7.3.5 Contributions to the SNR improvement 136
7.4 Conclusion 138
7.5 Acknowledgements 141
7.6 Appendix A: Compatibility with other gravitational wave (GW) search pipelines 142

7.7 Appendix B: Low-Latency Mode 143

Bibliography **145**

List of Figures

1.1 This figure shows the effect of a gravitational wave passing perpendicular to the page on a ring of test particles, for the plus polarization (left) and cross polarization (right). The ring alternately gets deformed into one of the two ellipses for each half period of the gravitational wave. The ellipses of the two polarizations are separated by an angle of $\frac{\pi}{4}$. This figure is taken from [1]. 3

1.2 This figure shows a simplified schematic for interferometric detectors. Light from the laser is split at the beamsplitter, causing it to travel down both arms, with a phase shift in one. The pair of test masses in both arms form a Fabry-Perot cavity, increasing the effecting arm lengths. In the absense of a gravitational wave, no interference pattern in produced. Whan a gravitational wave passes perpendicular to the arms, it causes the relative arm lengths to change, and hence an interfeence pattern can be detected at the photodetector. This figure is taken from [2]. 7

1.3 This figure shows an example power spectral density (PSD) plot for the LIGO and Virgo detectors, from the third observing run. This figure is taken from [3]. 9

2.1 A schematic showing which files are selected for an online rank, from the online analysis. The rectangles represent the two types of data products created by the online matched filtering process. The rectangles at the bottom represent the trigger snapshot files, and those at the top represent the background snapshot files. Each snapshot file is 4 hours long. The rectangles colored green are the ones selected by the online rank. Since the trigger snapshot files are discrete, all those having an overlap with the online rank duration are selected. In contrast, since the background snapshot files are cumulative, the earliest one containing all the background data for the duration of the online rank, and the latest one containing of it are chosen. These two are then subtracted to produce a background file containing exactly the background data for the duration of the online rank, to the granularity of the 4 hour snapshots. This process is repeated for every template bin in the analysis. 19

2.2 The ratio of the sensitive volume-times of the online rank to that of a traditional offline analysis over the same period of time, calculated for different mass bins and at different FAR thresholds. The fact that the VT ratios for all mass bins are close to 1 across FAR thresholds tells us that the online rank method is very close in sensitivity to a traditional offline analysis. The 5% loss in VT comes from the fact that the online analysis dropped approximately that much data. The peaks and troughs in the BBH line are because of the small number of statistics in that mass bin. 23

2.3 The ratio of the sensitive volume-times of the online rank augmented with triggers and background data from the periods of time dropped by the online analysis to that of a traditional offline analysis over the same period of time. We see that the 5% loss in VT seen in Fig. 2.2 is recovered by adding the 5% of dropped data. This shows that the online rank method is exactly as sensitive as a traditional offline analysis. 24

2.4 Histograms of the fractional chirp mass accuracy for the online rank (top) and histograms of the fractional chirp mass accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at chirp mass recovery as a traditional offline analysis. 27

2.5 Histograms of the fractional total mass accuracy for the online rank (top) and histograms of the fractional total mass accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at total mass recovery as a traditional offline analysis. 27

2.6 Histograms of the recovered time accuracy for the online rank (top) and histograms of the recovered time accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at time recovery as a traditional offline analysis. 28

2.7 Histograms of the fractional SNR accuracy for the online rank (top) and histograms of the fractional SNR accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at SNR recovery as a traditional offline analysis. 28

3.1 This figure shows the ratio of the VT of the HLV AllSky analysis (denoted "RB" for its two checkerboards) to the VT of the HL AllSky analysis (denoted "EJ" for its two checkerboards). We see that for the two lowest mass bins, sensitivity increases by around 2% - 4%. 35

3.2 This figure shows the real part of the SNR time series obtained in noise from an Early Warning template when using an FFT length of 4 seconds. We can see that despite having a mean of 0 as expected, the standard deviation is smaller than 1, indicating faulty whitening. 38

3.3 This figure shows the real part of the SNR time series obtained in noise from an Early Warning template when using an FFT length of 8 seconds. We can see that both the mean and standard deviation take their expected values, showing that whitening s functioning properly. 39

4.1 AllSky templates and IMBH templates on the $\log(\mathrm{m}_1)$-$\log(\mathrm{m}_2)$ plane. Here, the orange dots with $m_1 \leq 200 M_\odot$ are AllSky templates and blue dots with $m_1 \geq 200 M_\odot$ are IMBH templates. 45

4.2 An example of a particular online job's list of dropped data segments. The dropped data segments at the very start of the plot are because of slightly different start times of the online and offline analysis, whereas the ones after that are because of online analysis having failed to analyze those times, either because the data for those times was dropped in order to keep up with incoming live data, or because the online job was not functioning at that time. We see that the amount of dropped data is not too large, but the segments are dispersed throughout the period of the analysis. [4] shows that a typical online analysis drops around 5% of the total data. This data can be filtered offline in order to augment the online rank's results. This particular plot was made using a GstLAL online analysis that participated in the mock data challenge [5]. 50

4.3 This plot shows the results of two IMBH searches: one which only considers triggers with two or more detectors contributing to it, and another with no such restriction. The plot also shows the noise LR statistics of the former search for reference. We can see that the likelihood ratio (LR) statistics of the candidates of the search that only processes triggers with two or more detectors is well behaved, whereas the other one is not. This is because the IMBH search, if allowed to process single-detector triggers recovers an overwhelming amount of them, and the LRs (and hence FARs) of single-detector triggers are difficult to accurately calculate. As a result, for O4, GstLAL's IMBH search doesn't process single-detector triggers. 52

4.4 Example of $\rho - \xi^2$ noise model for one of the low-mass template groups collected for Laser Interferometer Gravitational-Wave Observatory (LIGO) Hanford detector during the mock data campaign described in [5] with the lightblue dashed line as a $\xi^2 = 1$ contour 65

4.5 2D $\rho - \xi^2$ noise model sliced at the $\xi^2 = 1$ contour and projected onto the ρ dimension. The noise model with the extrapolation closely follows the expected distribution in black curve at $\rho > 9$ as opposed to the red curve without the extrapolation. 65

4.6 Effect of applying the new extinction model on the noise LR histogram. The effects of candidate clustering and differing relative contributions of template bins to the candidate set are modeled by modifying the noise LR histogram, n(L) to $A[1 - e^{-cn(L)}]$. The values of A and c are found by curve fitting the modified noise LR histogram to the candidate LR histogram. We can see that this process is effective from the fact that the noise LR histogram with the new extinction extinction is very close to the candidate LR histogram in the noise region (low LR). 66

4.7 VT ratios of the fourth observing run (O4) search with the third observing run (O3) search. The solid lines represent the full O4 search as compared to the O3 search, whereas the dashed lines represent the O4 search without the extrapolation in $\rho-\xi^2$ noise model as compared to the O3 search. Since the extrapolation removes false positives from the candidates, it corrects previous mis-estimations of the VT done in the O3 search, superficially lowering the O4 VT. The VT ratios of the two lowest mass bins are slightly less than 1. This is because the O3 temlpate bank sampled the lower mass parameter space with a higher minimum match (0.99) as compared to O4 (0.97). Additionally, we see that the highest mass bin has a 50% - 100% increase in sensitivity, arising from the LR improvements and the additional IMBH search done in O4. 67

4.8 This plot shows the ratio of VT of the O4 combined AllSky+IMBH search to that of the O4 AllSky search, and it shows us how much sensitivity we gain by combining the IMBH and AllSky searches, as compared to just the AllSky search. As expected the sensitivities of the 3 lowest mass bins are unchanged, since the IMBH bank is not sensitive in that region. The VT of the IMBH bin increases by 6% - 7%. 68

4.9 VT ratio of the O4 search with the noise model extrapolation as compared to without. The extrapolation removes false positives from the set of candidates, and in the process also downweights a small amount of real GW signals. The former effect corrects the overestimate of VT that happens without the extrapolation, whereas the latter decreases the real sensitivity of the search. Both these effects contribute to the lower than 1 VT ratio seen here. This effect is mostly seen in the BNS ans NSBH regions, whose VT goes down by 10% - 15%. The BBH and IMBH regions are mostly unaffected. 69

5.1 An example of an event (GW200129_065458) having templates with high match in multiple template bins. Bin 818 has the best match with the GW candidate, and recovers it in both Hanford and Livingston as a coincidence. Bin 838 has a lower match than bin 818, causing it to recover the candidate as a Livingston single. This will lead to the candidate being added to the $\rho-\xi^2$ background histogram of bin 838, causing signal contamination for bin 838. This is shown in Fig. 5.2. The events passing the ρ and ξ^2 constraints, and hence recorded by the Background Filter are outlined in orange. 74

5.2 An example of signal contamination in a $\rho - \xi^2$ histogram for Livingston. The contamination can be seen as a protrusion to the histogram at (ρ, ξ^2/ρ^2) $\sim$ (15, 0.004), a region ususally occupied exclusively by GW signals. This contamination was caused by GW200129_065458 being recovered as a single event in this template bin, which is not the best match bin for that GW candidate, as demonstrated in Fig. 5.1. Note that kernel smoothing has been applied to this histogram. 76

5.3 The ρ and ξ^2 constraints for recording events. The bottom right area bounded by the blue lines is the area in which the Background Filter records events. If the events also pass the time constraint, the user can choose to remove them from the $\rho - \xi^2$ histogram. The result of doing so, to remove the contamination caused by GW200129_065458 is also shown. The same histogram, without using the Background Filter, and hence with contamination is shown in Fig. 5.2. Note that kernel smoothing has been applied to this histogram. 77

5.4 The distribution of component masses of the blind injection set, colored by injected ρ. Blind injections are used to replicate the contamination caused by GW signals in the data. 80

5.5 The decrease in VT caused by signal contamination due to the presence of blind injections in the data. The two highest mass bins are the most affected. The presence of GW signals will also have a similar effect. . . . 82

5.6 The VT of the rerank, which has blind injections with the Background Filter applied, compared to that of the control run, which has neither. The fact that all four lines are close to 1 tells us that the Background Filter is successful in removing nearly all of the contamination caused by the presence of the blind injections in the data. The peaks and dips in the highest mass bin curve are explained by the smaller number of injections in this bin as compared to other bins, leading to greater variance. 83

5.7 The VT with the Background Filter recording events with looser constraints, as compared to the VT with the Background Filter recording events with the regular constraints. The fact that the VT ratios for all four mass bins are close to 1 shows that loosening the constraints does not improve sensitivity. The peaks and dips in the highest mass bin curve are explained by the smaller number of injections in this bin as compared to other bins, leading to greater variance. Both analyses included the 868 blind injections described in Sec. 7.3. The regular constraints are described in Sec. 5.3, whereas the looser constraints are described in Appendix 5.7. 86

5.8 The VT with the Background Filter removing contamination from all blind injections, as compared to the VT with the Background Filter removing contamination from blind injections with FAR $\leq$ 1 per 5 months. The first represents the best-case performance of the Background Filter, whereas the second represents the current (O4) performance, designed to prevent noise events from being removed from the background. This graph shows that the current system recovers almost all of the lost sensitivity due to signal contamination. 87

5.9 The average bank correlation function of a BNS template in descending order of template match, as compared to that of an IMBH template, calculated for the five closest template bins. Since IMBH templates correlate well with other IMBH templates across template bins, an IMBH GW signal will be recovered by multiple template bins, increasing the probability of signal contamination. This is not the case for BNS template bins, and it is more likely a BNS GW signal will be recovered by only one template bin, resulting in fewer cases of signal contamination. 89

5.10 The autocorrelation function of a BNS template, as compared to that of an IMBH template. Since there are multiple peaks in autocorrelation function of the IMBH template, a quiet IMBH GW signal could be recovered in different detectors at different times, corresponding to the different peaks in the IMBH autocorrelation function. This will result in the GW singal being recovered as multiple single detector events rather than a single coincident event, which leads to signal contamination. Since the BNS autocorrelation function does not have multiple peaks, signal contamination is less likely for BNS template bins. 91

6.1 Triggers found in one month's data for Hanford (top) and Livingston (bottom), colored by their log-likelihood ratio. All the bright points to the right of the boundary are known gravitational wave candidates, and all those to the left of the boundary are glitches, and so not considered gravitational wave candidates, and not used for background estimation. 104

6.2 Distribution of component masses as measured at Earth for the BBH simulation set. 108

6.3 Cumulative histograms of our search results as a function of likelihood ratio. The orange line represents the corresponding histogram expected from noise during the same time frame. 109

6.4 Distribution of injected SNRs for recovered injections above $\mathcal{L} = 35(\sqrt{2\mathcal{L}} = 8.37)$. Missed injections with network SNR above 12 and detector SNRs greater than 7 (indicated by the shaded contour) are discussed in the appendix. 110

6.5 Efficiency of recovering injections at different injected SNRs 111

6.6 An example of a Q-transform plot showing a glitch in Livingston, causing a simultaneous injection to be missed 115

7.1 This figure shows two examples of sky maps. The one on the top is relatively well constrained in terms of sky location, while the one of the bottom is less constrained. Louder GW signals being detected in more number of detectors produces more constrained, and hence better sky maps. These sky maps were produced using the BAYESTAR package on simulated GW signals. 118

7.2 This figure shows a schematic of the template splitting process enabled by the manifold rectangle structure. Each rectangle has a template at its center, and the metric is calculated at that location. The boundaries of the rectangle represent the area (or volume, depending on the dimensionality of the space) occupied by the template. This schematic shows a template splitting into two new templates, one of which again splits into two new templates, and again one more time. This schematic shows templates only in the m_1-m_2 space. For the fourth observing run, the SNR Optimizer operates in the m_1-m_2-χ_{eff} space. 123

7.3 This figure shows the χ^2 value plotted against the $dt - d\phi - dSNR$ term of the LR implemented by GstLAL, for various triggers. The two quantities, while not perfectly correlated, are well correlated, and selecting the χ^2 = 3 threshold for calling a coincidence physical is almost equivalent to selecting a threshold value for $dt - d\phi - dSNR$. The tiny values of χ^2 seen on the left side of the plot are numerical noise in the calculation of χ^2, and actually represent a value of 0. 127

7.4 This plot shows a histogram of the χ^2 value for two types of triggers. The first, represented in blue, are simulated GW signals, and are guaranteed to be physical. The second, represented in orange, are triggers with randomly drawn times, and are highly unlikely to be physical. We see that the $\chi^2 = 3$ threshold serves as a perfect discriminator between physical and unphysical triggers. The tiny values of χ^2 for the injection triggers are numerical noise in the calculation of χ^2, and actually represent a value of 0. 128

7.5 A histogram of the combined signal-to-noise ratio (SNR) improvement calculated from the SNR Optimizer uploads as compared to GstLAL uploads. 131

7.6 This plot shows the complementary cumulative distribution of mean percent improvement in SNR due to the SNR Optimizer vs SNR (top) and inverse false alarm rate (FAR) (bottom). In both cases, the mean improvement goes down slightly with an increase in SNR or inverse FAR 132

7.7 This plot shows the complementary cumulative distribution of average percent of times the SNR Optimizer finds a higher SNR than GstLAL (i.e. it is the preferred event) for a candidate vs SNR (top) and inverse FAR (bottom). In both cases, the average preferred event percent goes down slightly with an increase in SNR or inverse FAR 133

7.8 This plot shows cumulative histograms of the 90% sky area for GstLAL and the SNR Optimizer. We see that sky maps produced from the SNR Optimizer triggers are on average more constrained than those produced from GstLAL triggers. In other words, the SNR Optimizer results are more precise than those of GstLAL. 134

7.9 This plot shows cumulative histograms of the searched area for GstLAL and the SNR Optimizer. We see that sky maps produced from the SNR Optimizer triggers on average have a lower searched area than those produced from GstLAL triggers. This means we have to go to lower confidence levels for the SNR Optimizer for the sky map to exclude the true location of the trigger, as compared to GstLAL, implying that the SNR Optimizer results are more accurate than those of GstLAL. 135

7.10 This plot shows cumulative histograms of the searched probability for GstLAL and the SNR Optimizer. The SNR Optimizer line is closer to the diagonal than the GstLAL line. This means that the higher precision and accuracy of the SNR Optimzier is also self-consistent. This plot is commonly called a P-P plot. 136

7.11 The end-to-end latencies (top) and internal latencies (bottom) of the SNR Optimizer. The SNR Optimizer has an internal timeout of 240 seconds, causing the internal latencies to be cut off at that value, and the end-to-end latencies to only rarely exceed that. 137

7.12 This plot shows the contributions from finding a better template (top), and real-time template whitening (bottom), towards the overall SNR gain obtained by the SNR Optimizer. During the process of finding a better template, the templates with the maximum SNR is selected, and hence the contribution from that is strictly positive. Real-time template whitening is only guaranteed to produce positive contributions *on average*. For example, since the PSD can never be perfect due to non-stationary components in the noise, using the PSD measured in real-time to whiten the templates will not always produce a higher SNR, but will do so on average. Fig. 7.13 shows the remaining contributions to SNR gain. . . . 139

7.13 This plot is a continuation of Fig. 7.12. This plot shows the contributions from using a higher FFT length (top), and higher data sampling rate (bottom) towards the overall SNR gain obtained by the SNR Optimizer. Similar to what is explained in Fig. 7.12, the higher FFT lenght and data sampling rate are only guaranteed to produce positive contributions *on average*. 140

List of Tables

2.1 The candidate list of the online rank. The first five candidates correspond to the previously reported events of GW200129_065458, GW200115_042309, GW200128_022011, GW200208_130117, and GW200209_085452. However, the times are different than those reported in [6], because the data was shifted in time. The candidates and parameters reported by the online rank are identical to those reported by the traditional offline analysis in Tab. 2.2 25

2.2 The candidate list of the offline analysis. The first five candidates correspond to the previously reported events of GW200129_065458, GW200115_042309, GW200128_022011, GW200208_130117, and GW200209_085452. However, the times are different than those reported in [6], because the data was shifted in time. The candidates and parameters reported by the online rank in Tab. 2.1 are identical to those reported by the traditional offline analysis here 26

3.1 This table shows the 30 candidates that only GstLAL found significantly in low-latency, during O4b. Out of these, 19 events were found only by GstLAL, with no contribution from other search pipelines. 36

4.1 Parameters of the GstLAL O4 AllSky template bank. 44

4.2 Parameters of the GstLAL O4 IMBH template bank. 46

4.3 The candidate list for the offline search using GstLAL's O4 methods. The search is run over a week of O3 data. The first, second, third, fourth, fifth, and seventh candidates in this list correspond to the GWs GW190521_074359, GW190519_153544, GW190521, GW190513_205428, GW190517_055101, and GW190514_065416, previously reported in [7,8]. The remaining 4 candidates all arise from the IMBH search. Given the uncertainty surrounding the IMBH space of GW signals, we make no claims regarding the origin of these 4 candidates 59

4.4 The candidate list for the offline search using GstLAL's O3 methods. The search is run over a week of O3 data. The first, second, third, fourth, fifth, and eighth candidates in this list correspond to the GWs GW190521_074359, GW190513_205428, GW190519_153544, GW190517_055101, GW190521, and GW190514_065416, previously reported in [7, 8]. 60

6.1 Candidate gravitational wave events with the 10 smallest false alarm rates and largest SNRs. The first seven triggers as well as the last one correspond to known gravitational wave candidates: GW170817, GW170814, GW170809, GW170823, GW170729, GW170817a, GW170818, and GW170727. The other two triggers have not been previously reported as gravitational wave candidates. 112

Acknowledgments

This dissertation is dedicated to my parents, Meenal and Yogen. I owe all of my success to them. I would additionally like to thank my sister, Maithilee, and brother-in-law, Jan for always being there for me; my cousins, Advait, Akshay, Priyanka; my family, Atya, Kaka, Mama, Mami and my grandparents for giving me so much love; and all my friends, from school and undergrad life for getting me through all the tough times. Finally, I would like to thank my advisor, Chad, for being the best advisor I could have asked for; and all of the GstLAL group, for being a wonderful group of people. I have been blessed to be surrounded by incredible people all of my life.

This material is based upon work supported by the National Science Foundation under Award No. NSF-2310912, NSF-2308881, NSF-2311355, NSF-2201445, NSF-2110594, NSF-2103662, NSF-2011865, NSF-2018299, NSF-1934752. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Chapter 1
Introduction

This work is structured as follows: In Chapter 1, an introduction to gravitational waves is provided. Chapter 2 contains an introduction to the GstLAL gravitational wave search pipeline, while also describing a new method adopted by GstLAL that outsources the matched filtering procedure for an offline analysis to an online analysis run over the same data. Chapter 3 contains details about the GstLAL's online operations during the fourth observing run, whereas Chapter 4 describes GstLAL's offline operations for the fourth observing run using the method previously described on these online results. Chapter 5 talks about a method for preventing gravitational wave signals from being included in the background used to assign significance to gravitational wave candidates. Chapter 6 contains a description of a new search pipeline called MASS, which relies on the parameter space metric to dynamically create new gravitational wave tempalte rather than rely on a fixed bank of them, and Chapter 7 describes the MASS pipeline being modified to create a targeted, hierarchical search to follow-up on GstLAL's candaiates in an online search to facilitate better skymaps, and hence multi-messenger efforts.

1.1 Gravitational Wave Formalism in General Relativity

Gravitational waves, first predicted by Albert Einstein in 1916 as a consequence of general relativity, are ripples in the fabric of spacetime produced by non-spherically symmetric motion of massive objects. Unlike classical Newtonian gravity, which describes gravitational interactions as instantaneous forces, general relativity formulates gravity as the curvature of spacetime caused by mass and energy distributions. When massive objects undergo asymmetric acceleration—such as in binary mergers or stellar collapses—these distortions propagate outward as gravitational waves, traveling at the speed of light.

The mathematical formulation of gravitational waves arises naturally from Einstein's field equations, which describe the dynamic relationship between spacetime curvature and energy-momentum distribution [9, 10]:

$$R_{\mu\nu} - \frac{1}{2}Rg_{\mu\nu} = \frac{8\pi G}{c^4}T_{\mu\nu} \tag{1.1}$$

where $R_{\mu\nu}$ is the Ricci curvature tensor, R is the Ricci scalar, $g_{\mu\nu}$ is the metric tensor, and $T_{\mu\nu}$ represents the energy-momentum tensor of matter.

1.1.1 Linearized Field Equations and Wave Solutions

To describe gravitational waves, we consider the weak-field approximation, where the metric of spacetime deviates only slightly from the Minkowski metric $\eta_{\mu\nu}$. This deviation is represented as a perturbation $h_{\mu\nu}$:

$$g_{\mu\nu} = \eta_{\mu\nu} + h_{\mu\nu}, \quad |h_{\mu\nu}| \ll 1 \tag{1.2}$$

$$\eta_{\mu\nu} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.3}$$

Under this assumption, Einstein's field equations can be linearized, and in vacuum (where $T_{\mu\nu} = 0$), they reduce to the simple wave equation [9, 10]:

$$\Box \bar{h}_{\mu\nu} = 0 \tag{1.4}$$

where $\Box$ is the d'Alembertian operator, and $\bar{h}_{\mu\nu}$ is the trace-reversed perturbation.

$$\Box = \frac{\partial^2}{\partial t^2} - \nabla^2 \tag{1.5}$$

$$\bar{h}_{\mu\nu} = h_{\mu\nu} - \frac{1}{2}\eta_{\mu\nu}h \tag{1.6}$$

As commonly done for radiative solutions, we apply the Lorentz gauge [9, 10]:

$$\partial^\nu \bar{h}_{\mu\nu} = 0 \tag{1.7}$$

This equation admits plane wave solutions of the form:

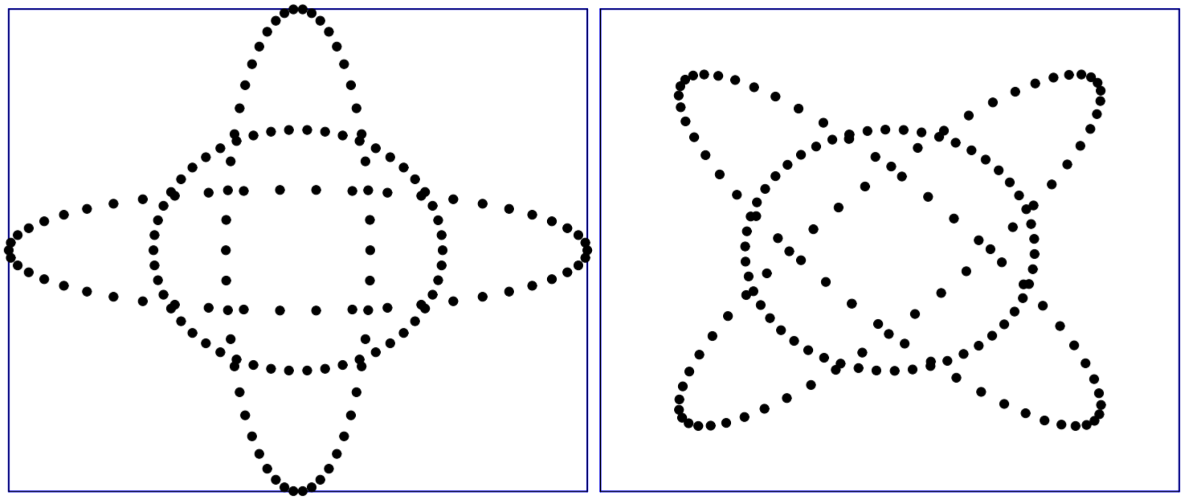


Figure 1.1. This figure shows the effect of a gravitational wave passing perpendicular to the page on a ring of test particles, for the plus polarization (left) and cross polarization (right). The ring alternately gets deformed into one of the two ellipses for each half period of the gravitational wave. The ellipses of the two polarizations are separated by an angle of $\frac{\pi}{4}$. This figure is taken from [1].

$$\bar{h}_{\mu\nu} = A_{\mu\nu} e^{i(k_\sigma x^\sigma)} \tag{1.8}$$

where k^σ is the wave vector, and $A_{\mu\nu}$ is the amplitude of the wave.

1.1.2 Transverse-Traceless Gauge and Polarization Modes

By adopting the transverse-traceless (TT) gauge, we eliminate unphysical degrees of freedom, reducing the number of independent components of $h_{\mu\nu}$ to two polarization states [1,9,10]:

$$h = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & h_+ & h_\times & 0 \\ 0 & h_\times & -h_+ & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{1.9}$$

These correspond to the “plus” (h_+) and “cross” ($h_\times$) polarizations, which affect spacetime in distinct ways. A passing gravitational wave distorts a ring of test particles into an oscillating elliptical shape, where the plus polarization alternately stretches and compresses space, and the cross polarization does the same at an angle $\frac{\pi}{4}$ from the plus polarization. This effect is shown in Fig. 1.1

1.1.3 Amplitude of Gravitational Waves

In this formalism, the amplitude of a gravitational wave from some mass distribution can be calculated to be [1,10]:

$$h_{ij} \approx \frac{2G}{c^4 r} \frac{d^2 Q_{ij}}{dt^2} \tag{1.10}$$

where Q_{ij} is the quadrupole moment of the mass distribution:

$$Q_{ij} = \int \rho x_i x_j d^3 x \tag{1.11}$$

This relation reveals two important features of gravitational waves:

1. The factor of $\frac{G}{c^4}$ tells us that the amplitude of gravitational waves is incredibly small, and only the most massive objects in the universe, astronomical bodies, can produce gravitational waves loud enough to be detected

2. The factor of $\frac{d^2 Q_{ij}}{dt^2}$ tells us that only mass distributions with a changing quadrupole moment can produce gravitational waves

1.2 Sources of Gravitational Waves

1.2.1 Compact Binary Coalescences (CBCs)

One of the primary sources of gravitational waves is the merger of compact binary systems, such as binary black holes (BBH), binary neutron stars (BNS), and neutron star-black hole (NSBH) pairs. These systems radiate gravitational waves as they inspiral due to energy loss, leading to an increasing frequency and amplitude—a characteristic "chirp" signal—until the final merger.

The strain amplitude observed at a distance D_L from the source is given by:

$$h \approx \frac{2G}{c^4} \frac{M v^2}{D_L} \tag{1.12}$$

where M is the total mass of the system, and v is the characteristic velocity of the inspiraling objects.

As the binary components orbit each other, they lose energy through gravitational wave emission, causing their separation to decrease and orbital velocity to increase. This results in a well-defined inspiral phase, followed by a highly dynamic merger phase and a final ringdown phase, where the remnant object settles into a stable state. The evolution of the gravitational waveform can be accurately modeled using the Post-Newtonian formalism for the inspiral phase.

The first direct detection of gravitational waves, GW150914, was attributed to a BBH merger, marking a pivotal moment in astrophysics. Since then, numerous CBC events have been detected, including GW170817, the first observed BNS merger, which was accompanied by electromagnetic counterparts, confirming the connection between BNS mergers and short gamma-ray bursts. The detection of such events provides valuable insights into fundamental physics, such as constraints on the equation of state of neutron stars and tests of general relativity in the strong-field regime.

Additionally, the measurement of parameters such as the chirp mass M_c, given by:

$$M_c = \frac{(m_1 m_2)^{3/5}}{(m_1 + m_2)^{1/5}} \tag{1.13}$$

is crucial for characterizing binary systems. The chirp mass directly influences the frequency evolution of the gravitational waves and is among the most accurately determined parameters from observed signals.

1.2.2 Core-Collapse Supernovae and Other Transient Sources

Massive stars undergoing core collapse at the end of their life cycles can also emit gravitational waves. If the collapse is asymmetric, significant variations in the quadrupole moment produce gravitational radiation. However, these signals are highly complex and challenging to model due to the turbulent, multi-dimensional nature of the explosion. Should the collapse result in the formation of a neutron star, additional gravitational waves may arise from rotational instabilities.

1.2.3 Continuous Gravitational Wave Sources

Certain astrophysical objects, particularly rapidly rotating neutron stars with asymmetries, emit nearly monochromatic gravitational waves over extended periods. If a neutron star possesses a non-axisymmetric deformation, often described as a "mountain", its rotation leads to continuous gravitational wave emission. Detecting these signals offers insights into the internal structure and equation of state of neutron stars.

1.2.4 Stochastic Background of Gravitational Waves

Beyond individual sources, a stochastic background of gravitational waves arises from the cumulative superposition of numerous weak, unresolved signals across the universe. This background can originate from astrophysical events such as unresolved CBCs, or from primordial processes in the early universe, such as cosmic inflation or phase transitions. Detecting this background would provide groundbreaking insights into the conditions of the early universe and physics beyond the Standard Model.

1.3 Gravitational Wave Detectors

Gravitational wave detectors are designed to measure the minuscule distortions in spacetime caused by passing gravitational waves. The challenge in detecting these waves lies in their extremely small strain values, often on the order of $h \sim 10^{-21}$, necessitating highly sensitive instruments.

1.3.1 Interferometer-Based Detectors

Gravitational waves alter distances between objects in a characteristic manner that can be measured by interferometric techniques. The strain, defined as the fractional change in length, is given by:

$$h = \frac{\Delta L}{L} \tag{1.14}$$

where ΔL is the change in arm length due to the passing gravitational wave, and L is the original arm length. Modern ground-based gravitational wave detectors, such as LIGO, Virgo, and KAGRA, operate based on the principles of Michelson interferometry. These detectors consist of long perpendicular arms with highly reflective mirrors at their ends. A laser beam is split into two, traveling down each arm and reflecting back. The recombined beam produces an interference pattern that shifts when a gravitational wave passes through the detector, altering the relative arm lengths.

To improve sensitivity, these interferometers incorporate Fabry-Perot cavities within their arms. These cavities consist of additional highly reflective mirrors, allowing the laser light to bounce multiple times before exiting. This effectively increases the optical path length, amplifying the phase shift induced by a gravitational wave. Furthermore, power recycling techniques enhance laser intensity, improving the signal-to-noise ratio. A

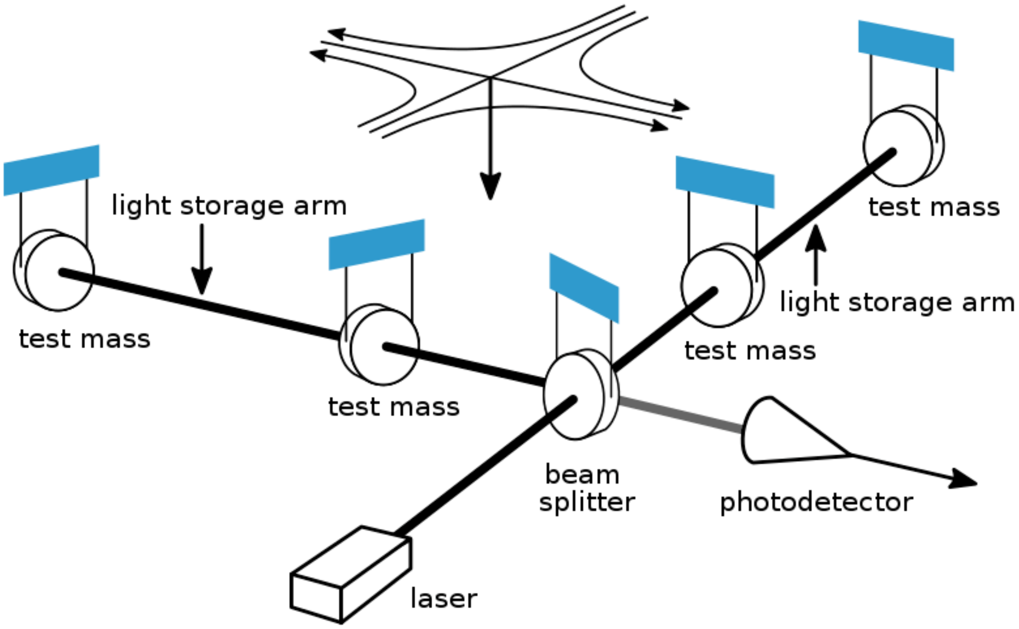


Figure 1.2. This figure shows a simplified schematic for interferometric detectors. Light from the laser is split at the beamsplitter, causing it to travel down both arms, with a phase shift in one. The pair of test masses in both arms form a Fabry-Perot cavity, increasing the effecting arm lengths. In the absense of a gravitational wave, no interference pattern in produced. Whan a gravitational wave passes perpendicular to the arms, it causes the relative arm lengths to change, and hence an interfeence pattern can be detected at the photodetector. This figure is taken from [2].

schematic for these detectors is shown in Fig. 1.2.

1.3.2 Noise Sources and Mitigation

Detecting gravitational waves requires overcoming various noise sources, which can mask weak signals. The dominant noise sources include:

- **Quantum Noise:** Arising from the quantum nature of light, it manifests as shot noise at high frequencies and radiation pressure noise at low frequencies. Quantum squeezing techniques have been employed to reduce this limitation.

- **Seismic Noise:** Vibrations from the Earth, including tectonic movements and human activity, affect the detector's stability. Advanced suspension systems isolate mirrors to mitigate these effects.

- **Thermal Noise:** Due to fluctuations in the material structure of mirrors and suspensions, thermal noise limits sensitivity at intermediate frequencies. Improvements in mirror coatings and cooling techniques help in reducing this noise.

- **Newtonian Noise:** Caused by fluctuating gravitational fields due to environmental mass movements, this noise is difficult to shield against and requires active monitoring and subtraction techniques.

The total noise floor of the detector is characterized by its power spectral density (PSD), $S_n(f)$, which quantifies the contribution of different noise sources to the measured signal:

$$\langle n(f)n^*(f')\rangle = \frac{1}{2}S_n(f)\delta(f-f') \tag{1.15}$$

An example of a PSD is shown in Fig. 1.3

1.3.3 Future Developments

Next-generation gravitational wave detectors aim to push sensitivity limits further. Planned projects include:

- **Einstein Telescope (ET):** A proposed underground detector with 10 km arms to improve low-frequency sensitivity.

- **LISA (Laser Interferometer Space Antenna):** A space-based interferometer designed to detect millihertz gravitational waves, enabling the study of supermassive black hole mergers and early universe signals.

- **Cosmic Explorer (CE):** A proposed U.S.-based detector with 40 km arms, significantly enhancing detection range and frequency coverage.

Advances in quantum optics, cryogenic technologies, and improved noise reduction techniques will play a key role in these future detectors.

1.3.4 Data Analysis and Signal Extraction

Even after mitigating noise sources, gravitational wave signals remain buried within residual detector noise. Sophisticated data analysis techniques need to be applied to extract signals. For modeled gravitational wave searches, the primary tool used is matched filtering, which correlates an a-priori known gravitational wave waveform template against the data. Many other data analysis techniques are used, and these will be explained in further chapters, particularly for the GstLAL search pipeline.

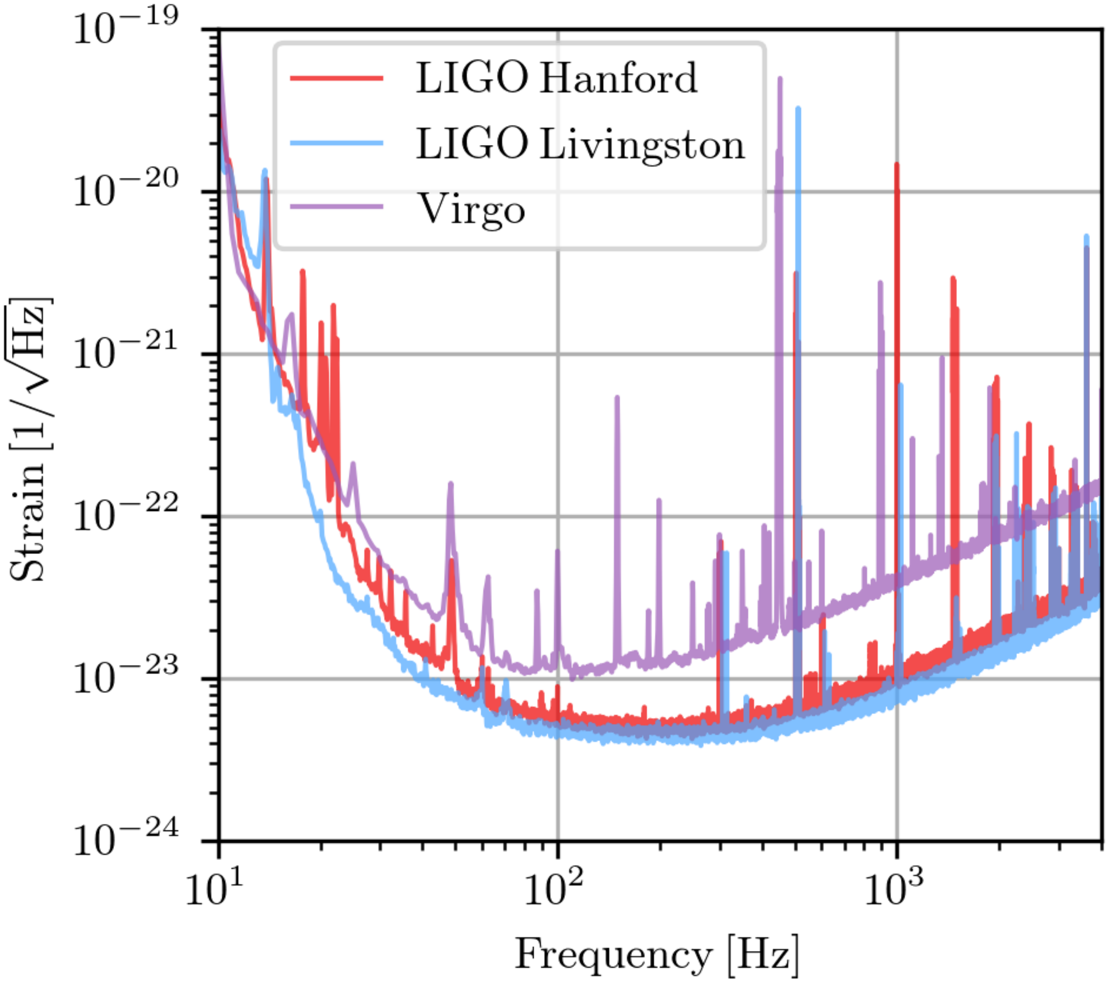


Figure 1.3. This figure shows an example power spectral density (PSD) plot for the LIGO and Virgo detectors, from the third observing run. This figure is taken from [3].

Chapter 2 | How Many Times Should We Matched Filter Gravitational Wave Data? A Comparison of GstLAL's Online and Offline Performance

This chapter is a reprint of [4]

Searches for gravitational waves from compact binary coalescences employ a process called matched filtering, in which gravitational wave strain data is cross-correlated against a bank of waveform templates. Data from every observing run of the LIGO, Virgo, and KAGRA collaboration is typically analyzed in this way twice, first in a low-latency mode in which gravitational wave candidates are identified in near-real time, and later in a high-latency mode. Such high-latency analyses have traditionally been considered more sensitive, since background data from the full observing run is available for assigning significance to all candidates, as well as more robust, since they do not need to worry about keeping up with live data. In this work, we present a novel technique to use the matched filtering data products from a low-latency analysis and re-process them by assigning significances in a high-latency way, effectively removing the need to perform matched filtering a second time. To demonstrate the efficacy of our method, we analyze 38 days of LIGO and Virgo data from the third observing run (O3) using the GstLAL pipeline, and show that our method is as sensitive and reliable as a traditional high-latency analysis. Since matched filtering represents the vast majority of computing time for a traditional analysis, our method greatly reduces the time and computational burden required to produce the same results as traditional high-latency analysis. Consequently, it has already been adopted by GstLAL for the fourth observing run (O4) of the LIGO, Virgo, and KAGRA collaboration.

2.1 Introduction

Ever since the first observing run (O1) of the LIGO Scientific [11], Virgo [12], and KAGRA [13] collaboration, the field of gravitational-wave (GW) astronomy has proven to be an invaluable tool for probing the universe. By detecting mergers of black holes and neutron stars [6–8, 14], GW astronomy has given us the ability to study the universe in new ways. This has led to a host of new scientific results [15–17]. GW searches are the first step to producing results within GW astronomy.

Only GWs produced by the mergers of the largest compact objects in the universe like black holes and neutron stars are loud enough to be observable by GW detectors like the Laser Interferometer Gravitational-wave Observatory (LIGO), Virgo, and KAGRA. Even then, GW signals reaching Earth are very faint, and heavily dominated by detector noise. Matched filtering [18] is the primary tool employed by modeled GW searches to detect GW signals in noisy data. In this process, the data is corss-correlated against a template of a GW waveform predicted by general relativity, producing a signal-to-noise ratio (SNR) timeseries as output.

GstLAL [19–22] is a stream-based GW search pipeline that has contributed to the LVK's GW detections since O1. It implements time-domain matched filtering to recognize periods of time where GW signals are possibly buried in noise (called "triggers"). It then calculates a likelihood ratio (LR) [23–25] as a ranking statistic for assigning significance to these triggers. The triggers with particularly high LRs are retained and called GW candidates. Based on the LRs and rate of triggers recognized as noise, the LRs of candidates are converted to a false alarm rate (FAR), which represents our confidence in the candidate. Other search pipelines, such as PyCBC [26–28], MBTA [29, 30], SPIIR [31, 32], and IAS [33, 34], also use similar techniques.

The GstLAL pipeline can operate in one of two modes: a low-latency "online" mode, or a high-latency "offline" mode. The online mode is designed to matched-filter the data, produce triggers, recognize candidates, and assign FARs in near-real time. The results are then immediately uploaded to the Gravitational Wave Candidate Event Database (GraceDB) [35], from where a public alert can be sent if the upload meets certain criteria. The ability of GstLAL to produce results in near-real time and hence serve as an independent messenger in the detection of astronomical events is particularly useful. GW170817 [36, 37], a binary neutron star merger (BNS), is an excellent example of GW searches contributing to a multi-messenger detection, which led to many new scientific results [38, 39].

In contrast, the offline mode is designed to be run after all the data are available. The matched filtering, significance estimation, and FAR assignment stages are generally done one after the other in the offline analysis, and do not need to be done simultaneously like in the online analysis. Since the full background data can be used to assign significances to all candidates, it has traditionally been considered more sensitive than the online analysis. Since it operates in high latency, it is resilient to any processing delays, data availability delays, and hardware downtime. Consequently, it has traditionally been considered more reliable and robust than the online analysis. Because of this, the process of matched filtering, which is identical for both operating modes, has always been repeated for the offline analysis by all search pipelines, after it was initially done in low-latency for the online analysis. Since GW searches are expensive, both in terms of time and computational resources, this repitition has a significant human and computational cost. With GW searches always looking to produce more scientific results, they are becoming ever larger, expanding to new parameter spaces, and analyzing more data than ever before. Consequently, the associated cost of running them is quickly starting to become unfeasible.

In this work, we address whether the repitition of matched filtering, which requires the overwhelming majority of computational power and time of any GW search, is necessary. In Sec. 4.2, we describe the GstLAL pipeline and its two operating modes in detail. In Sec. 7.2, we introduce a novel technique in which the data products created by the matched filtering of an online analysis are used in an offline fashion. In Sec. 7.3, we compare the results of this technique to a traditional offline analysis, to answer the question of whether data needs to matched-filtered a second time.

2.2 Software

2.2.1 General GstLAL methods

The GstLAL workflow, in either operating mode, contains two broad stages: a setup stage, and a data processing stage. In the setup stage, input data products are precomputed for use during the data processing stage. Like all modeled GW searches, GstLAL uses a "bank" of GW waveform templates. The template bank ahead of O4 was generated using the `manifold` [40] software package as described in [41]. It covers waveforms produced by binary mergers with component masses from 1-200$M_\odot$ and dimensionless spins up to ±0.99. This results in a bank containing approximately 2 million total waveforms.

The template bank is divided into two equal halves, each of which individually covers the full parameter space described above, but with a lower template density than the full bank. This process is called checkerboarding. Its primary purpose is to divide the analysis across different computing clusters, so as to not have it depend on a single one. This is important for online operations, since it means even if there is instability at one computing cluster, the online analysis can still remain operational and produce low-latency results via the other cluster, albeit with a lower sensitivity, since a single checkerboard of the full bank is less dense than the full bank itself. According to [41], using a single checkerboard results in 1% lower matches (and hence SNRs) for 90% of GW signals as compared to using the full template bank.

Both checkerboards are split into ”template bins, each of around 1000 templates, sorted by linear combinations of their Post-Newtonian phase coefficients as described in [41]. The templates are additionally whitened using a power spectral density (PSD) that represents the frequency characteristics of detector noise in the data. The templates in a single template bin are processed together for the purpose of matched filtering and background collection, and consequently a single job within a GstLAL analysis corresponds to a specific template bin.

The next stage is the data processing stage. This involves the matched filtering process, significance assignment, FAR calculation, and uploading the results (the last only applicable for the online mode). The SNR timeseries produced by matched-filtering the data with a particular template is defined as:

$$\mathrm{SNR}(t) = \int_{-\infty}^{\infty} d\tau \hat{d}(t+\tau)\hat{h}(\tau) \tag{2.1}$$

where

$$\hat{d}(\tau) = \int_{-\infty}^{\infty} df \frac{\tilde{d}(f)}{\sqrt{S_n(|f|)/2}} e^{2\pi i f \tau} \tag{2.2}$$

is the data whitened with the single-sided PSD, $S_n(f)$, and

$$\hat{h}(\tau) = \int_{-\infty}^{\infty} df \frac{\tilde{h}(f)}{\sqrt{S_n(|f|)/2}} e^{2\pi i f \tau} \tag{2.3}$$

is the similarly whitened template. Since matched filtering needs to be performed for the full data for every template, it is extremely computationally intensive. Though dependent on factors like cluster availability and computational power of the processors, we can calculate a rough estimate of the percent of time in an offline analysis taken up by matched filtering. The duration of the first half of O4 is around 8 months. An offline analysis over this period of time would take around 2 months. The setup stage, combined with significance assignment and FAR calculation takes around 1 day. That means matched filtering accounts for more than 98% of the time required for an offline analysis.

The SNR timeseries of every template is used to form triggers, by identifying times when the SNR exceeds the threshold value of 4. Since GW signals are expected to be correlated across detectors, unlike noise, triggers that are formed using a single detector (also called a "single trigger") during a time when more than one detector was producing data (also called "coincident time") are assumed to have originated from noise, and are added to the background data. Additional techniques are also employed to ensure GW signals do not enter the background [42]. The background data thus collected is then used to rank the triggers using the likelihood ratio as described in [23]. Triggers with a high LR are retained as GW candidates, and the LRs of noise triggers, as well as the livetime of the analysis is used to convert the LRs of candidates into FARs.

In the following subsections, we will discuss how the implementation of these steps differs for the online and offline operating modes of the GstLAL analysis.

2.2.2 Online GstLAL Analysis

The online analysis is designed to ingest the data coming from the detectors in real-time, and produce results with minimal delay. Typically, GW candidates are uploaded to GraceDB within 10-20 seconds of the GW reaching Earth [43]. To facilitate this, the online analysis needs to ensure the following two principles are observed:

1. Causality: To process a GW candidate, the analysis can only use data available up to that point in time. It cannot wait for more data to become available in the future.

2. Keeping up with live data: The analysis cannot fall behind the incoming data. If it does, it needs to drop some data to catch up.

Currently, the GstLAL analysis has the ability to measure the PSD of the data in small batches of 4 seconds, and whiten the data accordingly (see Eq. (7.1)), thus effectively ensuring causality well enough to serve the near-real time search. However, this ability does not extend to template whitening (see Eq. (7.2)), and the templates need to be whitened during the setup stage. As a result, the online analysis uses a PSD projected to represent future noise, to whiten the templates. To minimize this effect, every week in O4, the GstLAL team has been whitening the templates used by the online analysis using the PSD measured over the previous week's data. The expectation is that this captures any changes in the noise characteristics with a timescale equal to or larger than a week.

Additionally, to ensure causality, in order to rank a particular trigger, the online analysis can only use background data collected up to the point of that trigger. It cannot wait for the full background data to be collected. As a result, it may happen that the background used to rank a specific trigger has not converged fully, and so the LR and FAR assigned to that trigger are not as accurate as they would have been had we waited for the full background to be accumulated. Hence, the online analysis is traditionally considered to be less reliable.

Finally, to ensure the principle of keeping up with live data, if for any reason the analysis is unable to process some stretch of data, that data is permanetly lost. The analysis cannot go back and re-analyze the data. Data drops like this generally happen for one of three reasons:

1. If the analysis hits a period of high latency, either due to its own internal processing, or due to external reasons like live data delivery failures, the analysis drops this data and moves on to newer data after waiting for a set amount of time (60 seconds).

2. During the running of the analysis, there will be regularly scheduled maintenance, both internally for the analysis, and externally for the hardware it is running on. It is attempted to make these periods of downtime as short as possible, and to make them coincide with periods of time when the detectors are not producing data, which might not always be the case.

3. Unintentional hardware downtime

Due to these reasons, the online analysis is traditionally considered to be less sensitive and robust as compared to an offline analysis.

While running an online analysis, if the FAR of a candidate crosses a certain threshold [44], it gets uploaded to GraceDB. If certain other criteria are met, a skymap is generated showing the likely sky location of the source of the GW [45–48], and a public alert is issued [49]. Additional information, such as low-latency parameter estimation of the source of the candidate [50], and the probability of astrophysical origin for different source classes [51–53] is also included in the public alert. Astronomers can then choose to follow up on this alert, and correlations can be made with other messengers [54–56]. In this way, the online analysis plays an instrumental role in multi-messenger astronomy.

2.2.3 Offline GstLAL Analysis

The offline analysis is designed to be robust, reliable, and more sensitive than the online analysis. The analysis is performed after the full data become available, and so there are no latency constraints. As a result, the analysis does not need to adhere to the principles of causality and keeping up with live data, like the online analysis did.

Consequently, the PSD used for template whitening can be directly measured from the full data itself, guaranteeing the best representaion of detector noise. In practice, the data are divided into week-long chunks, and the process of PSD measurement and template whitening is done separately for every chunk. This further improves how well we capture detector noise, and increases sensitivity.

Similarly, the filtering and ranking stages (which involves significance and FAR assignment), do not need to occur simultaneously for a given trigger in the offline analysis, and each stage can be done for all triggers before moving on to the next. This means that during the ranking stage, the background data from the full analysis can be used, leading to more reliable results, as well as an increase in sensitivity. With this in mind, the matched filtering stage and the ranking stage of the GstLAL offline workflow are completely modular, and designed to be run independently.

Additionally, the analysis does not need to drop data if it gets affected by some source of latency, whether it is a latency in its internal data processing or a disruption in data delivery. This means that the analysis is guaranteed to process all available data without dropping any like the online analysis, making it more reliable and sensitive. Furthermore, the analysis can make use of more robust data quality and data veto information, provided by external high-latency tools. This is thought to make the analysis more sensitive by removing obviously bad data that would have otherewise created false positives in both the candidates and the background.

The offline analsyis is commonly run with simulated gravitational wave signals injected into the data. These are called “injections”, and they are used to measure the response of the analysis (i.e. the sensitivity) for GWs from sources in various parameter spaces, at different distances, etc. Such offline analyses are used to generate more in-depth results, such as studies on population properties of compact binary systems [57, 58], and tests on general relativity [39, 59–61].

2.3 Methodology

In the descriptions of the online and offline GstLAL analyses above, the only differences between the matched filtering stages of the two are PSDs used for template whitening, and the fact that the two might not analyze exactly the same set of data. Since we expect the projected PSD used by the online analysis to still be a good approximation of the true PSD measured over the data that the offline analysis uses, the effect of PSD mismatch should be low [19]. Additionally, the weekly whitening of the online analysis templates using the previous week's PSD will lower any SNR loss due to PSD mismatch further. Similarly, with improvements to the stability of the online analysis [43], data distribution, and computing hardware done before O4, we expect the effect of data drops to also be low. In Sec. 7.3 we show that the online analysis only drops around 5% data as compared to the offline analysis, and also discuss ways to make up this lost 5%.

2.3.1 Online Rank

Based on this, we developed a novel technique that takes the data products created by the matched filter stage of an online analysis (i.e. triggers and background data), and replaces the offline analysis' matched filter stage with these. The rest of the offline analysis (i.e. the ranking stage), is kept the same. This is possible since the two stages are designed to be modular, as described in Sec. 4.2. We call this technique an "online rank", since the matched filtering is taken from the online analysis, and an offline ranking stage is added to it.

Alongside the modularity of the GstLAL offline workflow, the key to making an online rank possible is a feature of the GstLAL online analysis, called "snapshotting". Every 4 hours, every job in the the online analysis (each corresponding to a template bin) will write a snaphot of the triggers and background data it has collected to disk. The trigger snapshot files are discrete, i.e. each file will only contain triggers created since the previous snapshot, which amounts to 4 hours of triggers. In contrast, the background snapshot files are cumulative, i.e. each file will contain background data since the start of the analysis to the time of the snapshot. Snapshotting can also be used to save the progress of the online analysis in case something goes wrong and the analysis needs to be restored to a working state.

The setup process for an online rank involves going through all the trigger and background snapshot files written by the online analysis, and picking the relevant files to forward to the rank stage. The user can specify a start and end time for the online rank, which defines the duration of the online rank. Since trigger snapshot files are discrete, every trigger file whose duration (which is encoded in the filename) has an overlap with the duration of the online rank is forwarded to the rank stage. For the background snapshot files however, since they are cumulative, the earliest snapshot file that contains all the background data of the duration of the online rank is chosen. If the start time of the online analysis is different from the start time of the online rank, the latest background snapshot file that doesn't overlap at all with the duration of the online rank is also chosen. This is subtracted from the earlier file, to produce a background file that exactly contains the background data for the duration of the online rank, to the granularity of the 4 hour snapshots. This procedure is illustrated in Fig. 2.1. Typical of an offline analysis, this background file containing the full background data for the duration of the online rank is used to rank every trigger, leading to more reliable and sensitive results. Since trigger and background files are processed separately for every template bin, this process is repeated for every template bin. In this way, relevant files can be extracted from an online analysis that might be running for the full observing run, and offline results for a subset of the duration can be calculated from them. Since the snapshotting interval of 4 hours is relatively small as compared to typical analysis periods of many months, trigger and background data can be extracted from the online analysis for offline processing with high precision.

2.3.2 Offline Rank Stage Methods

After relevant trigger and background files for every template bin are chosen from the online analysis, they are forwarded to the offline rank stage. From this point on, the online rank analysis proceeds identically to the traditional offline analysis. In this, the trigger snapshot files are first aggregated into a single trigger file per template bin. Then, the background snapshot file for that template bin is used to assign significance to every trigger in the trigger file. These triggers files are "clustered" across template bins, a process in which only the most significant trigger in a window of 8 seconds is retained, leading to a set of GW candidates. This clustering process is designed to reduce the number of triggers that downstream processes need to process. The underlying assumption is that a single GW signal could create multiple triggers via multiple template bins, and by clustering, we are getting rid of these duplicates without losing any information, since

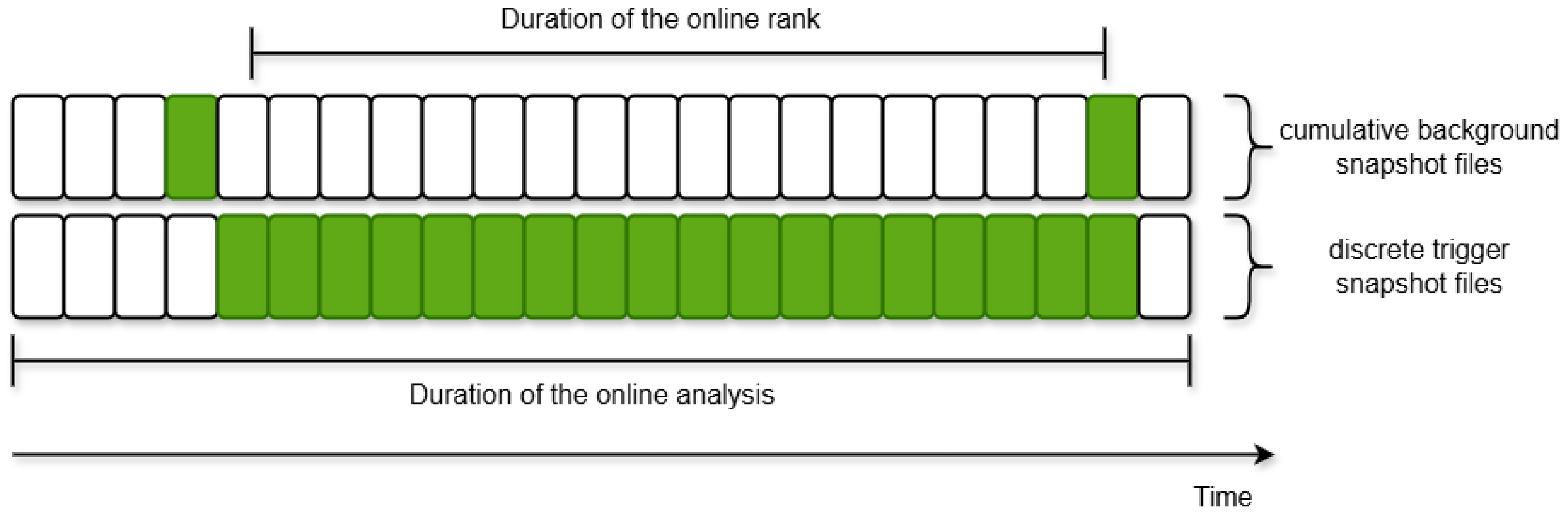

Duration of the online rank
cumulative background snapshot files
discrete trigger snapshot files
Duration of the online analysis
Time

Figure 2.1. A schematic showing which files are selected for an online rank, from the online analysis. The rectangles represent the two types of data products created by the online matched filtering process. The rectangles at the bottom represent the trigger snapshot files, and those at the top represent the background snapshot files. Each snapshot file is 4 hours long. The rectangles colored green are the ones selected by the online rank. Since the trigger snapshot files are discrete, all those having an overlap with the online rank duration are selected. In contrast, since the background snapshot files are cumulative, the earliest one containing all the background data for the duration of the online rank, and the latest one containing of it are chosen. These two are then subtracted to produce a background file containing exactly the background data for the duration of the online rank, to the granularity of the 4 hour snapshots. This process is repeated for every template bin in the analysis.

it's highly unlikely a single 8 second window could contain triggers from multiple GW signals, given the current detector sensitivities, and GW detection rates. The online analysis applies a similar clustering to the triggers within a template bin, but with a smaller window of 0.1 seconds.

Parallelly, the background snapshot file for every template bin is used to draw samples of noise triggers, and hence extrapolate the noise triggers to higher significances. These noise triggers then undergo a process called "extinction", in which the noise trigger LR statistics are modified to better resemble the triggers from that template bin. Specifically, two things are ensured:

1. the number of noise triggers from every template bin is similar to the number of triggers from that bin. In other words, each template bin must contribute equally to tbe background as well as foreground.

2. Even though the noise triggers don't explicitly undergo clustering, the effect of clustering can be modeled, and the LR statistics of the noise triggers are modified accordingly.

Extinction is applied to the noise triggers twice, the first time to each template bin-specific noise triggers, in order to model the effect of internal clustering applied by the online analysis, and the second time to the noise triggers aggregated across template bins, in order to model the effect of clustering applied when combining triggers across template bins as described above.

Finally, the extrapolated and extincted background and the livetime of the analysis is used to convert the LRs of candidates into FARs. If injections were involved, the FARs are used to determine which injections were found by the analysis and which were missed. This is in turn converted into a quantitative statement about the sensitivity of the analysis, using a quantity called the sensitive 4-dimensional volume-time that the analysis was able to observe, or the VT.

2.3.3 Dropped Data Refiltering

In order to make the online rank results even more reliable and sensitive, we can augment the inputs to the online rank with triggers and background data from times that the online analysis dropped. To do this, for every job in the online analysis, we take the list of times the job was functional and producing output. We then subtract that from the list of times a traditional offline analysis would have analyzed. This is called the segment list for the offline analysis, and it is calculated by external tools, by looking at the periods of time each detector was operational. This leaves us with a list of times that particular job has dropped. We calculate such a "dropped data segments" list for every job. Then, we can set up a traditional offline analysis using these dropped data segments. The only unusual aspect of this procedure is that while setting up this "dropped data refiltering" analysis, each job has its own segments list rather than a global one for every job.

By combining the online rank inputs with the results of the dropped data refiltering analysis, we can be sure the online rank produces offline results for exactly the same periods of time that the traditional offline analysis would have. The typical amount of dropped data for any job is around 5% of the total time covered by the offline segments.

2.3.4 Computational Cost Reduction

The online rank procedure requires an online analysis to have been run over the relevant period of data first. It enables us to re-use the matched filtering data products from the online analysis in order to get offline results. Since matched filtering is the bulk of the computational cost of any modeled GW search (98% as discussed in Sec. 4.2), over the course of an observing run (i.e. including running analyses to get both online and offline results), the online rank procedure represents approximately a 50% reduction in the total computational cose.

Given that the online analysis has alrady been run, we can calculate the time saved to get offline results via an online rank. Since the time required for matched filtering scales linearly with the amount of data analyzed, but the time required for an online rank does not strongly depend on the amount of data analyzed, the time saved because of the online rank method depends on the length of the analysis. The duration of the first half of O4 is around 8 months. The time required to perform a traditional offline analysis with injections over this period of time is approximately 4 months. We can get offline results for the same duration of time via an online rank in as low as 5 hours, if dropped data refiltering is not included. This represents a 99.8% reduction in the computational time in the best case scenario.

As discussed before, the typical amount of data dropped by an online analysis is 5%. This means that even if we choose to perform the dropped data refiltering analysis to augment the online rank, we still get approximately a 95% reduction in the amount of time required to get offline results.

2.4 Results

2.4.1 MDC Data Set and Analyses

In order to test the efficacy of our new method, we ran an online analysis over a mock data challenge (MDC). This involved running the analysis over 38 days of O3 data from the LIGO Hanford and Livingston detectors, as well as the Virgo detector. The data extended from 7 January 15:59:42 UTC 2020 to 14 February 20:39:42 UTC 2020. These data were then shifted in time by 125952000 seconds to extend from 4 January 10:39:42 UTC 2024 to 11 February 15:19:42 UTC 2024, to make them appear as though they were live data when we ran the online analysis. The MDC also involved an injection campaign. More details about the MDC, including details about the injection set used can be found in [5].

We then performed an online rank on this analyis, as well as a traditional offline analysis on the same amount of data. After accounting for the times when no detectors were producing data, we find that the online rank had a livetime of 34.41 days, whereas the offline analysis had a livetime of 36.05 days. This means that over the course of 38 days, the online analysis dropped around 4.5% of the data. To compensate for this, we also performed a dropped data refiltering analysis over the 4.5% dropped data. All of these analyses were performed using a single checkerboard (i.e. half of the full bank) of the GstLAL O4 template bank.

2.4.2 Sensitivity Comparisons

To compare the sensitivities of the online rank and the offline analysis, we can calculate the VTs of both, and then take the ratio of the two. Here, we have calculated the VT separately for injections with chirp mass in four different mass bins, roughly corresponding to four source categories: binary neutron star mergers (BNS, chirp mass between 0.5 to 2 $M_{\odot}$) neutron star-black hole mergers (NSBH, chirp mass between 2 to 4.5 $M_{\odot}$), binary black hole mergers (BBH, chirp mass between 4.5 to 45 $M_{\odot}$), and intermediate-mass black hole mergers (IMBH, chirp mass between 45 to 450 $M_{\odot}$). The VT is calculated at different FAR thresholds for considering an injection to be found by the analysis. The results of this VT comparison for different mass bins and FAR thresholds, for the pure online rank and offline analysis can be seen in Fig. 2.2. It shows us that the online rank is almost as sensitive as the offline analysis. We note that the 5% loss in the online rank VT as compared to the offline analysis lines up perfectly with the 5% of data dropped by

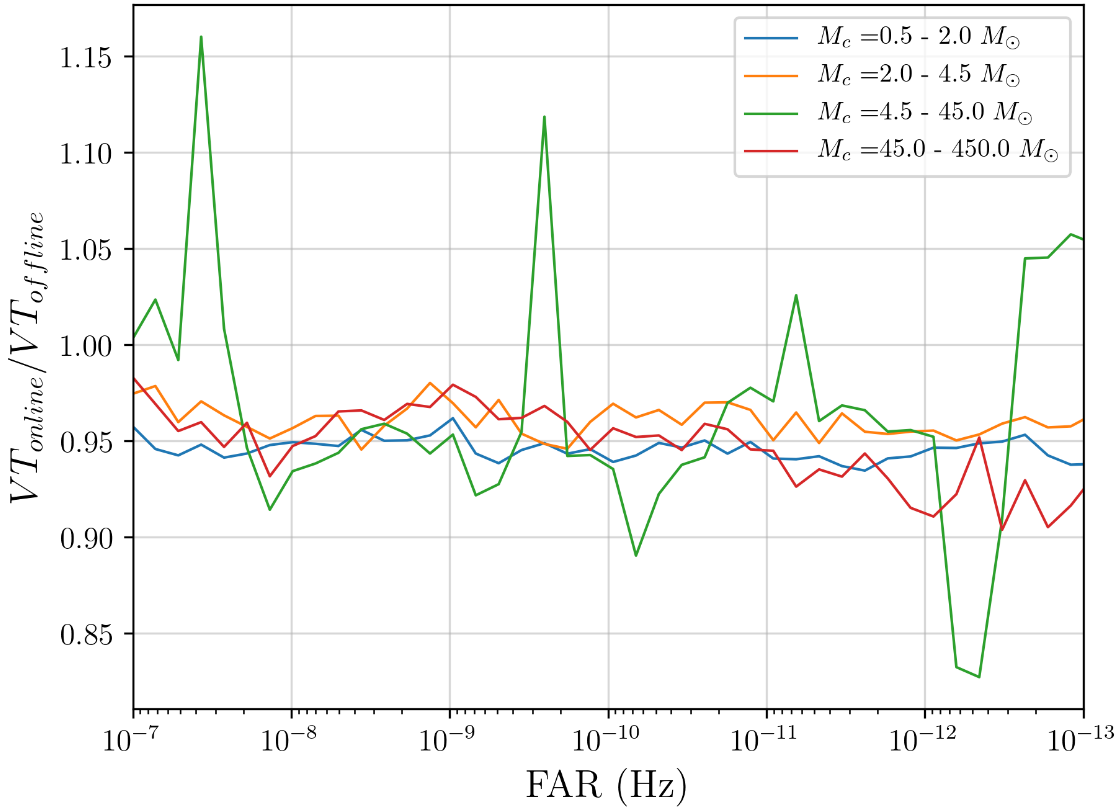


Figure 2.2. The ratio of the sensitive volume-times of the online rank to that of a traditional offline analysis over the same period of time, calculated for different mass bins and at different FAR thresholds. The fact that the VT ratios for all mass bins are close to 1 across FAR thresholds tells us that the online rank method is very close in sensitivity to a traditional offline analysis. The 5% loss in VT comes from the fact that the online analysis dropped approximately that much data. The peaks and troughs in the BBH line are because of the small number of statistics in that mass bin.

the online analysis.

Next, we repeat the procedure for the online rank augmented with the dropped data refiltering analysis, and compare its VT to the VT of the offline analysis. The result of this is shown in Fig. 2.3. The fact that the VT ratios are now much closer to 1 tells us that the previous 5% loss in VT was indeed coming from dropped data, and that by augmenting an online rank with a dropped data refiltering analysis, we can get offline results that are as sensitive as a traditional offline analysis in a fraction of the time.

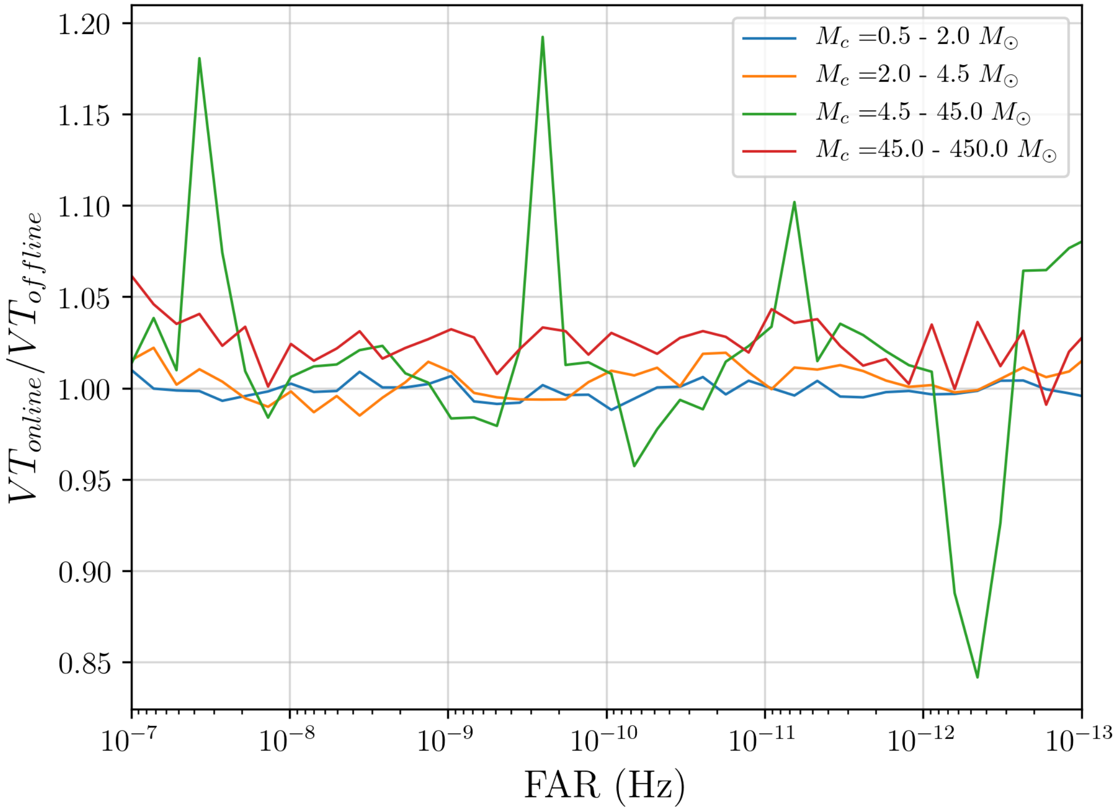


Figure 2.3. The ratio of the sensitive volume-times of the online rank augmented with triggers and background data from the periods of time dropped by the online analysis to that of a traditional offline analysis over the same period of time. We see that the 5% loss in VT seen in Fig. 2.2 is recovered by adding the 5% of dropped data. This shows that the online rank method is exactly as sensitive as a traditional offline analysis.

2.4.3 Candidate Lists

Next, we compare the candidate lists from the online rank augmented with dropped data refiltering and offline analysis as a further check on the reliability and sensitivity of the online rank. There are 9 previously reported GW signals in the MDC data, and both analyses are able to detect 5 of them with a FAR of 1/month (3.86×10^{-7} Hz) or less. Since the analyses were performed using only one checkerboard, we expect lower sensitivities than if they were performed with the full bank, and consequently we expect some of the previously known candidates to be missed by these analyses. The candidate list for the online rank is shown in Tab. 2.1, and that for the offline analysis is shown in Tab. 2.2. The top 10 candidates from both analyses are the same. Additionally, both

Rank	**FAR** (Hz)	**Time (UTC)**	$\mathbf{m_1}$ ($M_\odot$)	$\mathbf{m_2}$ ($M_\odot$)	$\mathbf{a_1}$	$\mathbf{a_2}$
1	5.45×10^{-34}	2024-01-26 01:34:58	40.86	30.5	0.05	0.05
2	1.43×10^{-13}	2024-01-11 23:03:09	5.24	1.77	-0.29	-0.29
3	6.58×10^{-13}	2024-01-24 21:00:11	59.52	57.08	0.17	0.17
4	1.15×10^{-12}	2024-02-05 07:41:17	50.36	34.57	-0.2	-0.2
5	8.69×10^{-9}	2024-02-06 03:34:52	50.36	40.86	-0.08	-0.08
6	1.20×10^{-7}	2024-02-04 08:49:54	176.4	184.0	0.6	0.6
7	4.16×10^{-7}	2024-01-26 06:22:45	70.35	79.75	0.45	0.45
8	9.95×10^{-7}	2024-01-17 21:57:48	79.75	59.52	-0.02	-0.02
9	1.2×10^{-6}	2024-02-06 03:40:07	40.86	42.6	0.73	0.73
10	1.46×10^{-6}	2024-01-29 08:21:54	126.3	57.08	-0.08	-0.08

Table 2.1. The candidate list of the online rank. The first five candidates correspond to the previously reported events of GW200129_065458, GW200115_042309, GW200128_022011, GW200208_130117, and GW200209_085452. However, the times are different than those reported in [6], because the data was shifted in time. The candidates and parameters reported by the online rank are identical to those reported by the traditional offline analysis in Tab. 2.2

analyses recover those 10 candidates with exactly the same template, as evidenced by the fact that they have the same primary and secondary masses (m_1 and m_2), as well as the same dimensionless spins (a_1 and a_2). Since the analyses were performed on O3 data shifted in time by 125952000 seconds, the reported times of the candidates do not match the times reported in the third Gravitational-Wave Transient Catalog [6]

2.4.4 Injection Parameter Recovery Comparisons

Since an injection campaign was conducted for both the online analysis (and hence the online rank), and the offline analysis, we can compare the parameters with which the online rank and offline analysis recovered injections. The FAR threshold used here for an injection to qualify as “found” is 1/month (3.86×10^{-7} Hz). The parameters we compare here are the chirp mass, total mass, time, and SNR. For this comparison, we calculate

Rank	**FAR** (Hz)	**Time (UTC)**	$\mathbf{m_1}$ ($M_\odot$)	$\mathbf{m_2}$ ($M_\odot$)	$\mathbf{a_1}$	$\mathbf{a_2}$
1	7.92×10^{-32}	2024-01-26 01:34:58	40.86	30.5	0.05	0.05
2	1.2×10^{-13}	2024-01-11 23:03:09	5.24	1.77	-0.29	-0.29
3	4.49×10^{-13}	2024-01-24 21:00:11	59.52	57.08	0.17	0.17
4	1.00×10^{-12}	2024-02-05 07:41:17	50.36	34.57	-0.2	-0.2
5	4.56×10^{-9}	2024-02-06 03:34:52	50.36	40.86	-0.08	-0.08
6	2.00×10^{-7}	2024-02-04 08:49:54	176.4	184.0	0.6	0.6
7	4.92×10^{-7}	2024-01-26 06:22:45	70.35	79.75	0.45	0.45
8	1.18×10^{-6}	2024-01-17 21:57:48	79.75	59.52	-0.02	-0.02
9	1.33×10^{-6}	2024-02-06 03:40:07	40.86	42.6	0.73	0.73
10	1.88×10^{-6}	2024-01-29 08:21:54	126.3	57.08	-0.08	-0.08

Table 2.2. The candidate list of the offline analysis. The first five candidates correspond to the previously reported events of GW200129_065458, GW200115_042309, GW200128_022011, GW200208_130117, and GW200209_085452. However, the times are different than those reported in [6], because the data was shifted in time. The candidates and parameters reported by the online rank in Tab. 2.1 are identical to those reported by the traditional offline analysis here

the fractional parameter accuracy of each found injection (i.e. $\frac{\text{recovered value}-\text{injected value}}{\text{injected value}}$), and add that value to the histograms of the detectors the recovered injection trigger comprised of. Time is treated slightly differently, in that we histogram the non-fractional accuracy (i.e. recovered time - injected time). We show the fractional accuracy for chirp mass, total mass, and SNR in Fig. 2.4, Fig. 2.5, and Fig. 2.7 respectively, and the time accuracy in Fig. 2.6. In all of these plots, the histograms for the online rank for each detector are plotted on the top panels, and the histograms for the offline analysis for each detector are plotted on the bottom panels. In each figure, we see that the shape, mean, and standard deviation of the histogram for the online rank is very similar to that of the offline analysis. This is further evidence that an online rank is very similar to a traditional offline analysis.

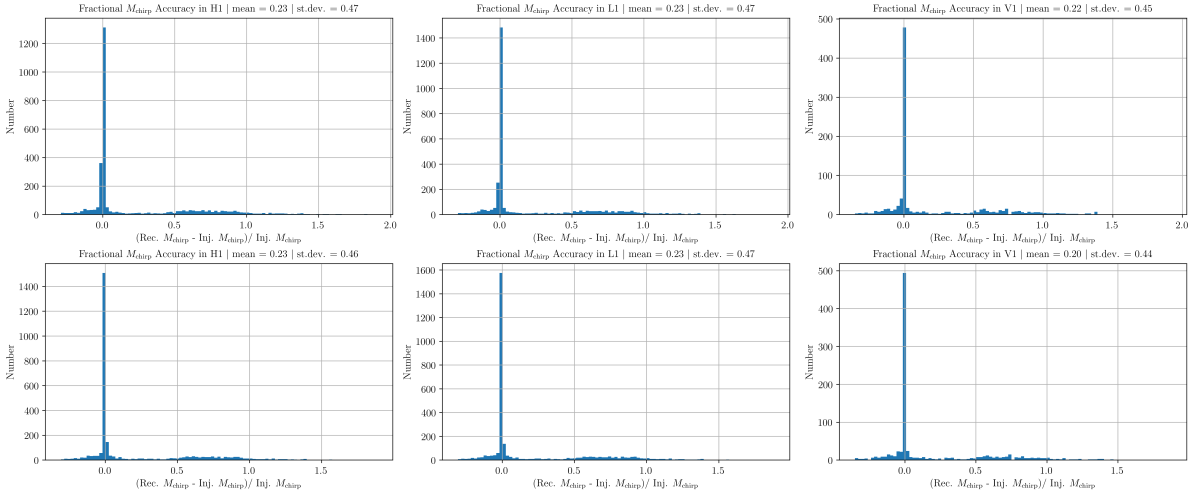


Figure 2.4. Histograms of the fractional chirp mass accuracy for the online rank (top) and histograms of the fractional chirp mass accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at chirp mass recovery as a traditional offline analysis.

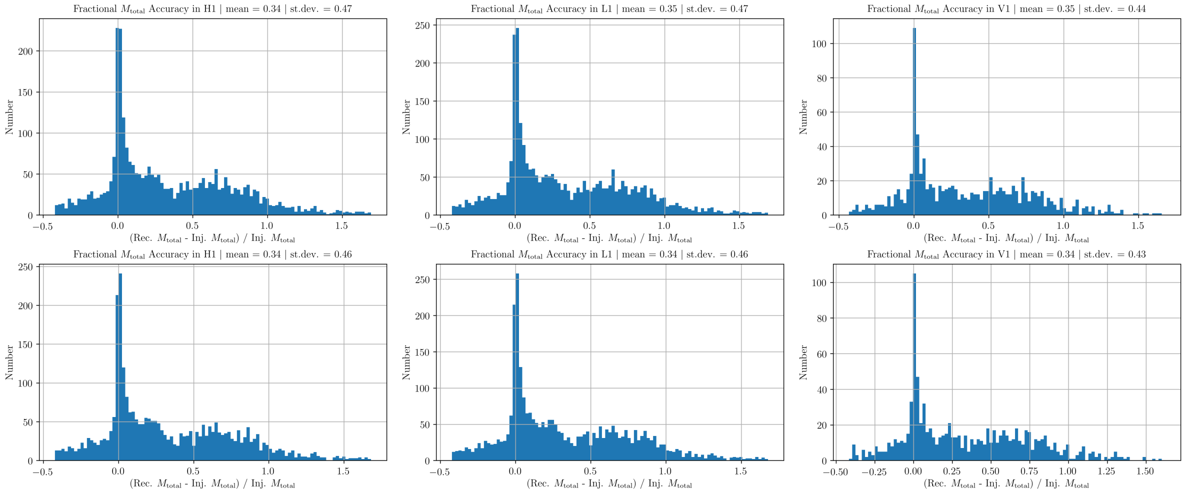


Figure 2.5. Histograms of the fractional total mass accuracy for the online rank (top) and histograms of the fractional total mass accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at total mass recovery as a traditional offline analysis.

2.5 Conclusion

In this work, we have decribed how a GstLAL analysis functions, and discussed the differences between the online and offline modes of operation. Both modes share a common matched filtering stage. The main differences between the two are that the online analysis needs to maintain causality to ensure low latencies, and that it needs to

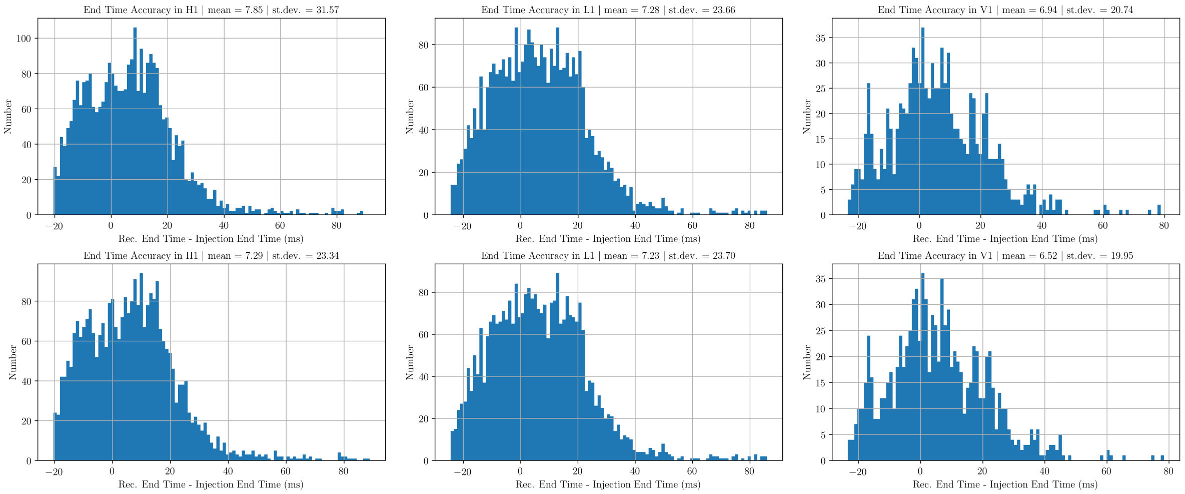


Figure 2.6. Histograms of the recovered time accuracy for the online rank (top) and histograms of the recovered time accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at time recovery as a traditional offline analysis.

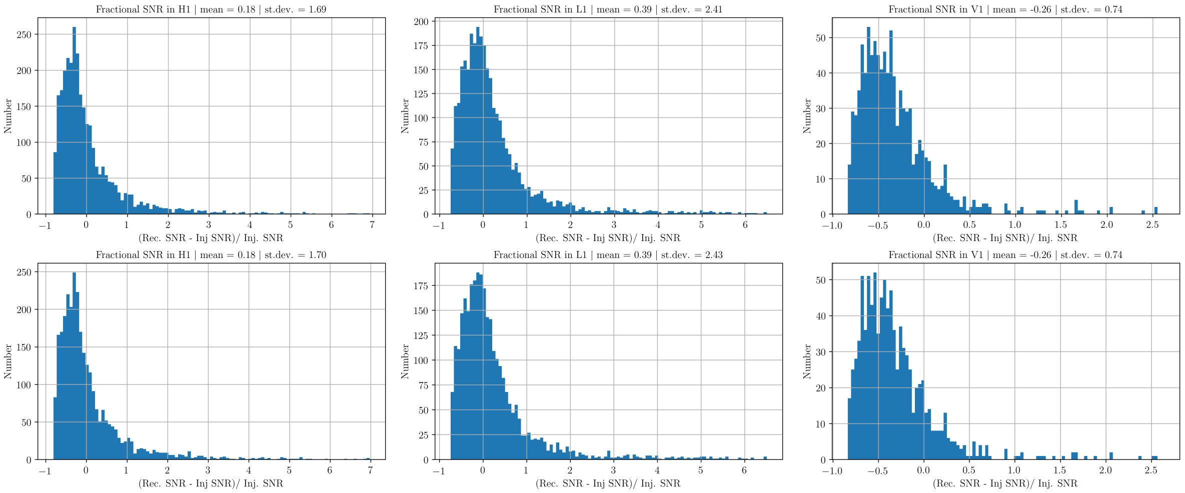


Figure 2.7. Histograms of the fractional SNR accuracy for the online rank (top) and histograms of the fractional SNR accuracy for the offline analysis (bottom). The histograms for the online rank and offline analysis are almost identical, showing that online ranks are just as good at SNR recovery as a traditional offline analysis.

keep up with live data. Due to the former, the online analysis cannot measure the PSD over the data after it becomes available, and it cannot use the full background data to rank triggers. Due to the latter, it needs to drop data if it starts lagging behind. Due to these reasons, the online analysis has traditionally been considered less sensitive and robust than the offline analysis.

We introduced a new method called an online rank, in which the data products created by the matched filtering stage of an online analysis (i.e. triggers and background data), which are saved as 4 hour snapshots of the online analysis, can be taken and processed in an offline fashion. This removes the drawback that the online analysis has, of not having the full background information available while ranking triggers. Since matched filtering takes up the large majority of time required for an offline analysis, by not repeating the process of matched filtering, and taking the matched filtering results from the online analysis instead, we can get reliable and sensitive offline results in a fraction of the time compared to what is required for a traditional offline analysis. Over the course of an observing run (i.e. including both online and offline results), this represents a 50% reduction in total computational cost.

Furthermore, we discussed a technique called dropped data refiltering in which the matched filter outputs of the online analysis are augmented by a small offline analysis which analyzes times dropped by the online analysis. This ensures an online rank analyzes exactly the same period of time as an offline analysis.

To test our method, we performed an online analysis on 38 days of LIGO and VIRGO O3 data. We found that the online analysis had dropped around 5% data as compared to the offline analysis, consequently suffering a 5% loss in VT. By adding the dropped data refiltering outputs to the online rank, we were able to show the online rank is exactly as sensitive and reliable as a traditional offline analysis.

Due to the significant reductions in computational effort and time enabled by online rank method, we belive the future of GW searches lies in this paradigm, in which the data is matched filtered only once per observing run, by the online analysis, and any other offline results that are needed will not have to repeat the matched filtering process again. For O4, the GstLAL group has already adopted the online rank method, enabling fast and reliable offline results, as well as fast testing on new development work.

2.6 Acknowledgements

This research has made use of data, software and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org/), a service of LIGO Laboratory, the LIGO Scientific Collaboration (LSC) and the Virgo Collaboration. We especially made heavy use of the LIGO Scientific, Virgo and KAGRA Collaboration (LVK) Algorithm Library. LIGO was constructed by the California Institute of Technology and the Massachusetts Institute of Technology with funding from the United States National Science Foundation (NSF) and operates under cooperative agreements PHYS-

0757058 and PHY-0823459. In addition, the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany supported the construction of Advanced Laser Interferometer Gravitational-Wave Observatory (aLIGO) and construction and operation of the GEO600 detector. Additional support for aLIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain.

The authors are grateful for computational resources provided by the the LIGO Lab culster at the LIGO Laboratory and supported by PHY-0757058 and PHY−0823459, the Pennsylvania State University's Institute for Computational and Data Sciences gravitational-wave cluster, and supported by OAC-2103662, PHY-2308881, PHY-2011865, OAC-2201445, OAC-2018299, and PHY-2207728. LT acknowledges support from the Nevada Center for Astrophysics. CH Acknowledges generous support from the Eberly College of Science, the Department of Physics, the Institute for Gravitation and the Cosmos, the Institute for Computational and Data Sciences, and the Freed Early Career Professorship.

Chapter 3
GstLAL's online operations during the fourth observing run

Chapter 2 contained an introduction to the GstLAL analysis [19–23, 43]. It also talked about its two modes of operation, the low-latency "online" mode, and the high-latency "offline" mode. It described how the online mode is important for multi-messenger efforts, and provided GW170817 [36, 37] as an example of the GstLAL online analysis contributing to a multi-messenger detection, leading to new scientific results [38, 39]. For every observing run of the LIGO Scientific, Virgo and KAGRA Collaboration (LVK), search pipelines like GstLAL analyze the full data in real time, in an effort to contribute to multi-messenger detections. In this chapter, we will talk about GstLAL's online operations in the fourth observing run (O4) of the LVK.

During O4, GstLAL has run three types of online searches:

1. AllSky search: This is a generic search for gravitational waves (GWs) arising from mergers of binary neutron stars (BNSs), neutron star-black hole binaries (NSBHs), and binary black holes (BBHs)

2. subsolar-mass (SSM) search: This is a search for GWs arising from compact binaries with at least one component below 1 $M_\odot$ [62].

3. Early Warning search: This is a search for GWs arising from mergers of BNSs. This search tries to detect the signal before the merger happens by using templates that are truncated before the merger phase, and hence give astronomers more time to point their telescopes at, and observe electromagnetically bright mergers. [63–65]

A detailed description of the GstLAL online workflow, including methods and working principles can be found in Chapter 2. Details of the template bank used for the O4 online

AllSky analysis can be found in Chapter 4 and [41]. Details about the O4 GstLAL online SSM template bank can be found in [66, 67], whereas details of the O4 GstLAL Early Warning template bank can be found in [68]. The AllSky and SSM template banks are both divided into two halves, with each half covering the full parameter space with a lower density than the full bank. This method is called checkerboarding. Each checkerboarded half of the template bank is run on a different computing cluster. This is done to ensure high availability of the online analysis. More details of the checkerboarding process can be found in Chapter 2.

3.1 O4a

The first half of the fourth observing run (O4a) took place from 24 May 2023 15:00:00 UTC to 16 January 2024 15:00:00 UTC [69]. Along with GstLAL, the modeled online searches MBTA [29, 30], PyCBC [26–28], and SPIIR [31, 32] were also running, as well as the unmodeled searches cWB [70, 71] and oLIB [72]. During O4a, the GstLAL team ran the AllSky, Early Warning, and SSM online searches using data from the LIGO Hanford and Livingston detectors.

My contributions to O4a were on the development side. I developed a method to remove GW signals from contaminating the background used to assign their significance. The details of this can be found in Chapter 5 and in [23, 42]. I also developed a method to get better skymaps from GstLAL search results, which is discussed in Chapter 7 and in [73]. I also contributed to operations, by maintaining the online analyses in the lead up to, and during O4a. The results of the test online analysis before O4a is discussed in detail in [43], and the results of GstLAL's O4a online operations are discussed in [66, 74, 75]. As a result, we will discuss them only briefly here. Finally, I was part of the analysis and paper writing team for GW230529, a neutron star–black hole binary (NSBH) detected during O4a.

3.1.1 Public Alerts

For the public alert threshold of a false alarm rate (FAR) of 1 per 5 months, there were 88 public alerts from various AllSky searches during this time. Out of these, 7 were determined to be false alarms, leaving us with 81 GW candidates. The GstLAL AllSky search contributed to 80 of these, and 26 were found significantly only by GstLAL [75]. There were no public alerts from any Early Warning searches, and SSM public alerts were disabled for O4a.

3.1.2 GW230529

GW230529 was a likely NSBH candidate detected in low latency on 2023 May 29 at 18:15:00 UTC in LIGO Livingston data by GstLAL, MBTA, and PyCBC analyses [76]. A public alert was sent out in near-real time [77], and a follow-up alert was sent out with updated parameter estimation and sky area information [78]. Only one detector was operational during the time of this candidtae, and being a single detector candidate, the significance was hard to esimate. After manual verification steps were performed by the analysis team, of which I was a part, the astrophysical origin of this candidate was established firmly. Due to its single detecotr nature, sky localization of the source of this candidate was not tight, giving a 90% credible sky area of 24,200 deg^2.

GW230529 was found by the GstLAL, MBTA, and PyCBC online analyses with an SNR of 11.3, 11.4, and 11.6 respectively. The online inverse FARs were 1.1, 1.1, and 160.4 years respectively. Offline follow-up studies found this candidate with better inverse FARs of 60.3 ,>1000, and >1000 years respectively.

Despite being a single detector candidate, GW230529 has contributed to many new scientific results. It's primary mass was estimated to be 3.6 $M_\odot$, and its secondary mass was estimated to be 1.4 $M_\odot$. The secondary can confidently be said to be a neutron star, however the primary could either be a black hole, or a neutron star. Additionally, it resides in the lower mass gap of 3 $M_\odot$ - 5 $M_\odot$, lending support to compact objects existing in this mass range. This has further implications on black hole and neutron star formation mechanisms. [76] also shows that GW230529 has significant implications on populations and merger rates. Specifically, it provides an update to the estimate of the merger rate of NSBHs, it shows that the inferred minimum mass of black holes in NSBH binaries decreases when GW230529 is included, and significantly increases the estimate of NSBHs expected to produce EM counterparts.

3.2 O4b

The second half of the fourth observing run (O4b) took place from 2024-04-10 17:00:00 UTC to 2025-01-28 17:00:00 UTC. For O4b, the GstLAL team launched and maintained an unprecedented 9 online analyses. These were:

1. AllSky HL checkerboard 1: "Edward"

2. AllSky HL checkerboard 2: "Jacob"

3. AllSky HLV checkerboard 1: "Bob"

4. AllSKy HLV checkerboard 2: "Rick"

5. Early Warning FFT length 4: "Alice"

6. Early Warning FFT length 8: "Alice"

7. SSM checkerboard 1: "Charlie"

8. SSM checkerboard 2: "Charlie"

9. An experimental analysis analyzing out-of-observing-mode data: "Esme"

We will discuss all of these analyses below

3.2.1 AllSky

3.2.1.1 HL vs HLV

The GstLAL team started O4b with 2 AllSky online analyses, each divided into 2 checkerboards. The Edward and Jacob analyses were checkerboards of an AllSky analysis that only analyzed data from the LIGO Hanford and Livingston detectors, whereas the Rick and Bob analyses were checkerboards of an AllSky analysis that analyzed data from the two LIGO detectors, as well as the Virgo detector. The reason for running both of these analyses parallelly at the start of O4b was that the effect of analyzing Virgo data was unknown. The two LIGO detectors had a sensitivity of around 150 Mpc, but the Virgo detector had a sensitivity of 50 Mpc, and it was unclear whether analyzing Virgo data would increase the overall sensitivity of the analysis.

After letting both analyses run for around 3 months, offline results were obtained from both of them by using the online rank method described in Chapter 2, and the sensitive volume-time i.e. VT of the two analyses was compared. The results of this are shown in Fig. 3.1. We see that the HLV analysis was around 2% - 4% more sensitive than the HL one for the two lower mass bins. As a result, the HLV analysis was chosen to be the production analysis for GstLAL for O4b, and the HL analysis was retired.

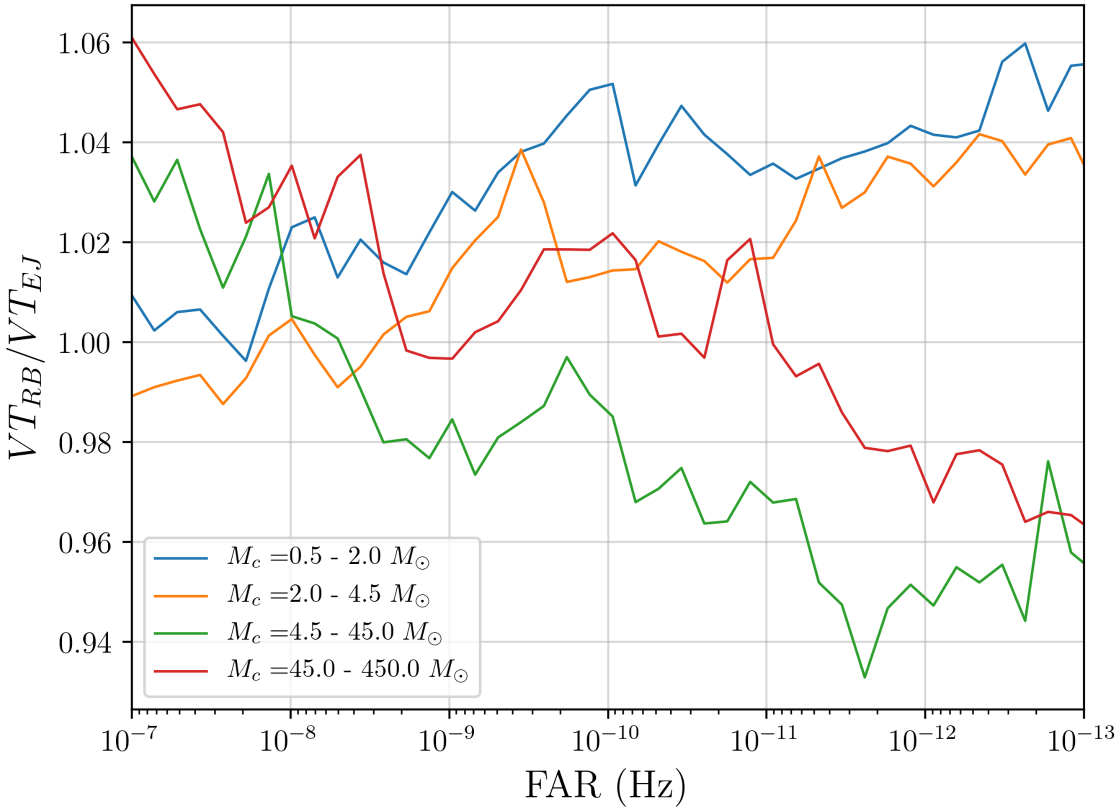


Figure 3.1. This figure shows the ratio of the VT of the HLV AllSky analysis (denoted "RB" for its two checkerboards) to the VT of the HL AllSky analysis (denoted "EJ" for its two checkerboards). We see that for the two lowest mass bins, sensitivity increases by around 2% - 4%.

3.2.1.2 Results

During O4b, there were 103 public alerts from the various AllSky searches. Out of these, 5 were deemed to be false alarms, and hence were retracted. Out of the remaining 98 events, GstLAL participated in 97 of them. Out of these, 30 events had only GstLAL contributing a significant trigger. These 30 events are shown in Tab. 3.1 Additionally, 19 of these events had contribution only from GstLAL, with no other pipelines detecting the candidate, even with low significance.

Superevent ID	**FAR** (Hz)	**Detectors**	$\mathbf{m_1}$ ($M_\odot$)	$\mathbf{m_2}$ ($M_\odot$)	**GCN**
S250109bi	2.049e-08	H1,V1	60.77	15.95	[79]
S241230ev	2.196e-08	H1,L1,V1	98.28	57.08	[80]
S241230bd	6.213e-14	L1,V1	59.51	50.35	[81]
S241225c	1.999e-15	L1,V1	13.42	9.53	[82]
S241210d	4.390e-08	H1,L1	76.48	79.74	[83]
S241201ac	4.321e-08	H1,L1,V1	98.28	79.74	[84]
S241122a	6.393e-08	L1	149.27	149.27	[85]
S241114bi	1.780e-13	L1,V1	14.44	8.15	[86]
S241113p	1.470e-14	L1,V1	23.48	18.27	[87]
S241110br	4.68e-09	H1,L1,V1	19.35	8.79	[88]
S241109p	6.041e-11	L1,V1	48.29	54.74	[89]
S241101ee	1.374e-11	L1,V1	64.70	20.93	[90]
S241011k	2.531e-34	H1,V1	21.94	5.89	[91]
S241009em	2.817e-09	H1,L1,V1	42.60	42.60	[92]
S241009l	3.310e-08	H1,L1,V1	59.51	57.08	[93]
S241002e	1.402e-21	L1,V1	39.18	40.86	[94]
S240921cw	8.018e-10	L1,V1	45.12	11.30	[95]
S240917cb	5.401e-08	H1,L1	59.51	67.46	[96]
S240916ar	1.652e-08	H1,L1,V1	11.18	11.42	[97]
S240915bd	3.294e-14	H1,V1	20.18	5.88	[98]
S240813d	1.806e-18	L1,V1	109.10	31.79	[99]
S240813c	2.614e-09	L1,V1	17.85	8.13	[100]
S240807h	2.011e-11	L1,V1	28.19	5.35	[101]
S240716b	7.861e-16	L1,V1	46.31	48.29	[102]
S240703ad	1.184e-13	L1,V1	39.18	40.86	[103]
S240627by	1.207e-08	H1,L1,V1	18.32	7.71	[104]
S240525p	1.677e-08	H1,L1,V1	46.31	40.86	[105]
S240514c	9.619e-09	L1	64.70	48.29	[106]
S240428dr	2.109e-14	H1,V1	21.60	17.89	[107]
S240422ed	3.095e-13	H1,L1,V1	3.44	1.18	[108]

Table 3.1. This table shows the 30 candidates that only GstLAL found significantly in low-latency, during O4b. Out of these, 19 events were found only by GstLAL, with no contribution from other search pipelines.

3.2.2 Early Warning

3.2.2.1 Data whitening problem

In the middle of O4b, it was found that the GstLAL Early Warning analysis was producing triggers at a very low rate. Investigations into this revealed that the Early Warning analysis was producing lower SNRs than expected. All GstLAL analyses have a SNR threshold of 4 for a detector to contribute to a trigger. Because of the lower SNRs, exponentially lower number of detectors were passing the SNR 4 threshold, resulting in very few triggers, and hence a non-functional search. Though the root cause is as yet unknown, it was found that during O4b, data quality had changed as compared to any data before O4b, in that the noise in the lower frequency range had been reduced. The GstLAL PSD measurement and data whitening when using an FFT length of 4 seconds was not able to adequately compensate for this decrease in noise in the lower frequency bands.

Though this affects all GstLAL analyses, the frequency range with lower noise is very small as compared to the frequency range of templates of a search like AllSky (15 Hz - 1024 Hz). However, since the Early Warning templates are cut off in time before the merger takes place, their frequency is also cut off at a very low value. This value is template dependent, but a typical frequency range for an Early Warning tempalte is 14 Hz - 45 Hz. This lies exactly in the range of lower noise and faulty data whitening, leading to much lower SNRs in the Early Warning search. It was found that this faulty whitening did not manifest when using a FFT length of 8 seconds. This is demonstrated in Fig. 3.2 and Fig. 3.3, where it is shown that we get the expected SNR statistics in noise (mean = 0, standard deviation = 1) when using an FFT length of 8, but not when using an FFT length of 4. As a result, in the middle of O4b, a new GstLAL Early Warning analysis was set up using an FFT length of 8 seconds, and this was made the production analysis.

3.2.2.2 Results

There were no significant public alerts from the Early Warning search of any pipeline during O4b.

3.2.3 SSM

3.2.3.1 Results

There were no significant public alerts from the SSM search of any pipeline during O4b.

3.2.4 Esme

The LIGO and Virgo detectors are often producing data event though they are not in “observing mode”. This means that extra validation that the data is reasonable has

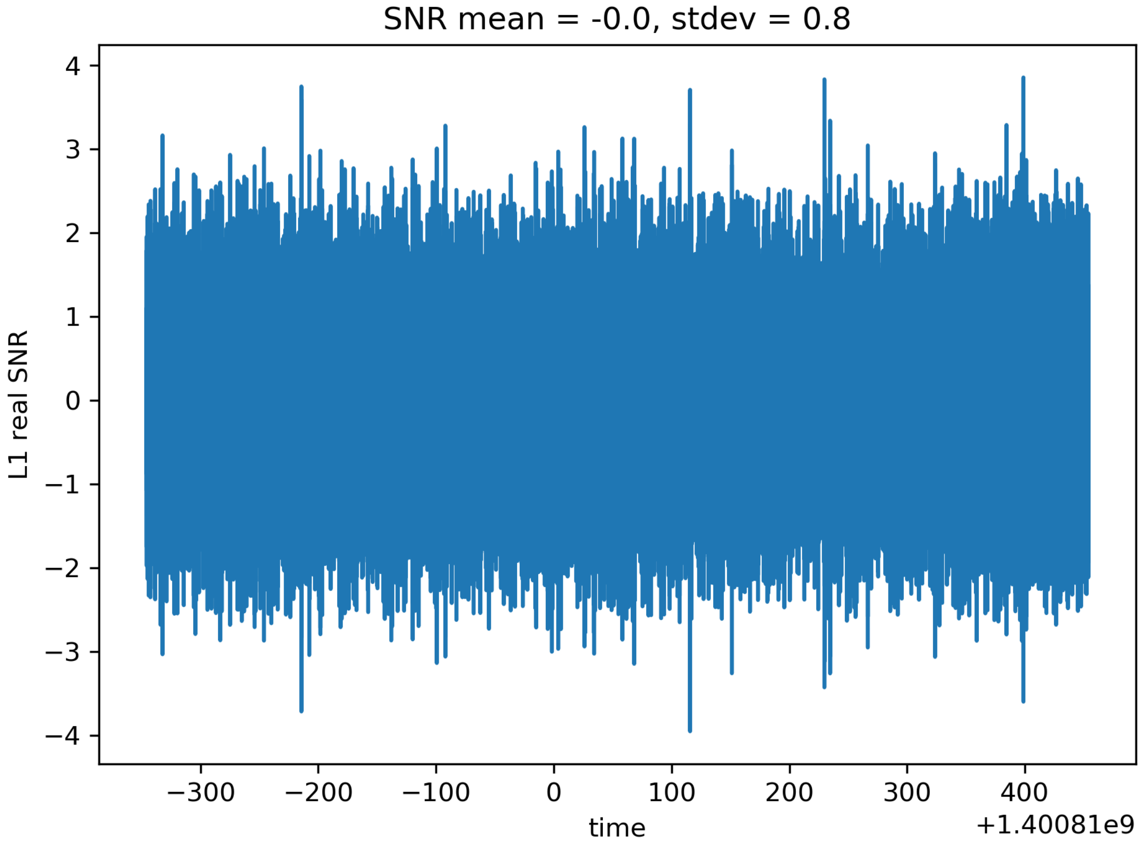


Figure 3.2. This figure shows the real part of the SNR time series obtained in noise from an Early Warning template when using an FFT length of 4 seconds. We can see that despite having a mean of 0 as expected, the standard deviation is smaller than 1, indicating faulty whitening.

not been done. Despite this, the data is available to be analyzed. Based on this, we developed a new method to analyze this out-of-observing-mode data, and launched an experimental analysis which we named “Esme" using that. This method selects which stretches of data to analyze not based on whether the detector is in observing mode, but rather it calculates the range of the detector at that time, and if it lies between certain values, then the data is analyzed. This is because if the detector is not producing reasonable output, the range of the detector will often be very low, or unreasonably high. The former happens when the detector is not running in a configuration allowing it to be sensitive to GW signals, and the latter happens when it is not tuned properly, and hence is full of glitches. The minimum and maximum range values were set to 10 and 600 Mpc for the Esme analsyis

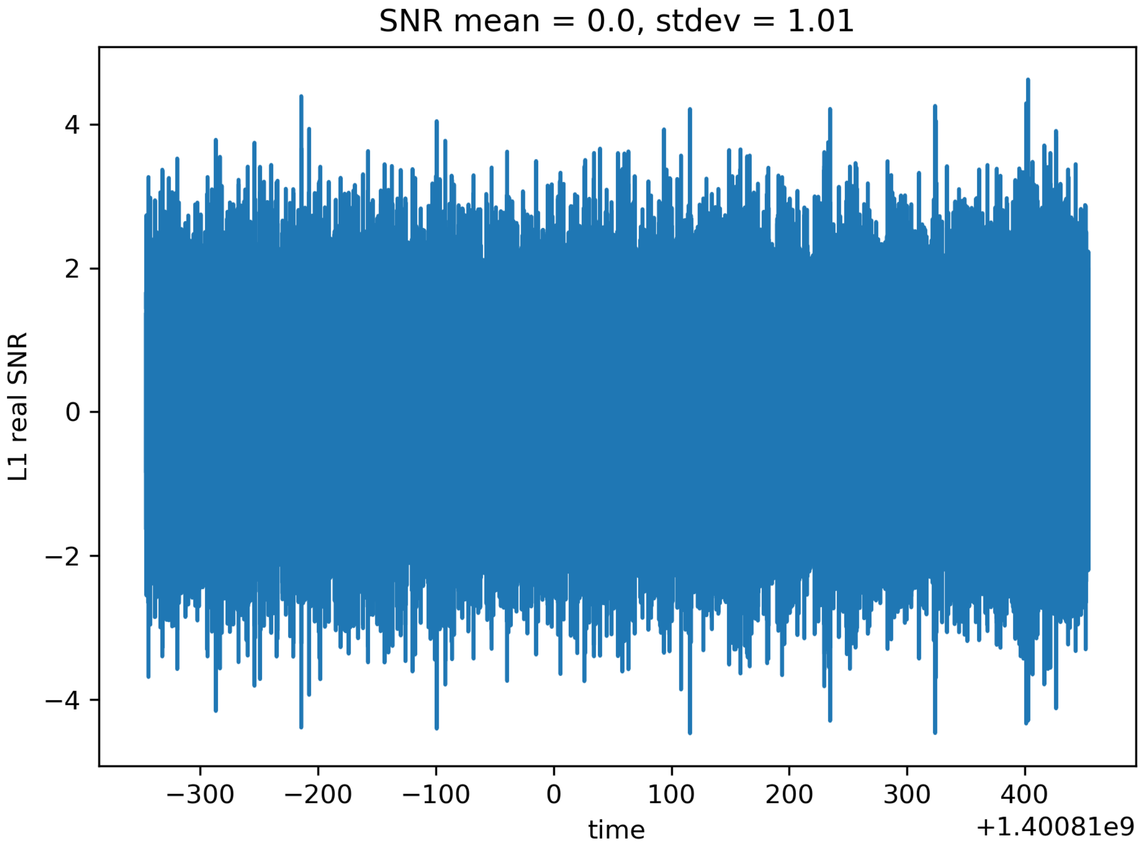


Figure 3.3. This figure shows the real part of the SNR time series obtained in noise from an Early Warning template when using an FFT length of 8 seconds. We can see that both the mean and standard deviation take their expected values, showing that whitening s functioning properly.

3.2.4.1 Results

The Esme analysis was running on O4b data, which is not public as of this time, and it also did not have public alerts enabled, since it was an experimental analysis. Hence, the results of this analysis cannot be discussed here.

3.3 Conclusion

O4b was a huge success for GstLAL's online operations. The AllSky analysis detected 97 GW candidates, out of which 30 candidates were found significantly only by GstLAL. Further, 19 of those were found only by GstLAL, with other pipelines not finding them at all. Many new features were implemented to increase sensitivity and reliability of results, and also to facilitate multi-messenger follow-up of candidates. Major problems with the Early Warning and SSM analysis were discovered and fixed.

Chapter 4 | New Methods for Offline GstLAL Analyses

In this work, we present new methods implemented in the GstLAL offline gravitational wave search. These include a technique to reuse the matched filtering data products from a GstLAL online analysis, which hugely reduces the time and computational resources required to obtain offline results; a technique to combine these results with a separate search for heavier black hole mergers, enabling detections from a larger set of gravitational wave sources; changes to the likelihood ratio which increases the sensitivity of the analysis; and two separate changes to the background estimation, allowing more precise significance estimation of gravitational wave candidates. Some of these methods increase the sensitivity of the analysis, whereas others correct previous mis-estimations of sensitivity by eliminating false positives. These methods have been adopted for GstLAL's offline results during the fourth observing run of LIGO, Virgo, and KAGRA (O4). To test these new methods, we perform an offline analysis over one chunk of O3 data, lasting from May 12 19:36:42 UTC 2019 to May 21 14:45:08 UTC 2019, and compare it with previous GstLAL results over the same period of time. We show that cumulatively these methods afford around a 50% - 100% increase in sensitivity in the highest mass space, while simultaneously increasing the reliability of results, and making them more reusable and computationally cheaper.

4.1 Introduction

By detecting gravitational waves (GWs) from the merger of compact objects like black holes and neutron stars, the LIGO Scientific, Virgo and KAGRA Collaboration (LVK) has revolutionized the field of gravitational wave astronomy. GW150914 was the first such detection of a GW [109], and since then close to 250 GWs have been detected [110]. While individual detections can often yield important scientific results, e.g., GW150914 [61], GW170817 [38,39], GW200105 and GW200115 [111], even richer scientific implication can be extracted from a collection of GW detections. To this end, the LVK releases *catalogs* of transient GW detections, called the Gravitational Wave Transient Catalog (GWTC). To date, four such catalogs have been released: GWTC-1 [14], GWTC-2 [7], GWTC-2.1 [8], and GWTC-3 [6]. These help inform results related to testing the theory of general relativity [16, 59, 112, 113], cosmology [15, 114], black hole properties and formation mechnisms [115–117], and binary population and merger rates [57, 58, 118].

These results are enabled by the detection of GWs by GW search pipelines. The contents of these catalogs are usually compiled from rigorous high-latency "offline" GW searches rather than low-latency "online" ones. GstLAL [19–22] is one such GW search pipeline. Like other modeled GW searches, it makes use of a bank of waveform templates predicted by general relativity. These templates are cross-correlated against the data in a process called matched filtering to calculate the signal-to-noise ratio (SNR). GstLAL performs matched filtering in the time domain [19, 21, 63] This process is used to identify periods of time possibly containing GW signals, which are called "triggers". Matched filtering is also used to inform the background data, against which triggers are ranked to evaluate their significance. A likelihood ratio (LR) [23, 24] is calculated as the ranking statistic. Triggers with a high LR are called GW candidates. The LRs of candidates are then compared with the LR statistics of background triggers, and using the livetime of the analysis, a false alarm rate (FAR) is calculated for every candidate. Some of these techniques are used by other GW search pipelines, such as IAS [33, 34], MBTA [29, 30], PyCBC [26–28], and SPIIR [31, 32].

This paper is structured as follows. In Sec. 4.2, we describe the GstLAL offline workflow. Sec. 7.2 is dedicated to describing the new features introduced in the GstLAL offline workflow in the lead up to, and during the fourth observing run (O4) of the LVK. Sec. 7.3 contains the results of the tests performed to evaluate these features, and to compare them against third observing run (O3) results.

4.2 Overview of the GstLAL Offline Analysis

The GstLAL offline analysis can be broadly divided into two stages, each of which is further divided into smaller stages. These are:

1. setup stage

    (a) template bank creation stage
    (b) power spectral density (PSD) measurement stage
    (c) singular value decomposition (SVD) of templates and template whitening stage

2. data processing stage
    (a) matched filtering stage
    (b) rank stage

4.2.1 Template bank creation stage

For O4, two GstLAL template banks, namely the AllSky and intermediate-mass black hole (IMBH) banks were generated using `manifold` [40, 119], which is a binary-tree approach to template bank generation.

4.2.1.1 AllSky template bank

The AllSky template bank includes templates in the binary neutron star (BNS), neutron star–black hole binary (NSBH), and binary black hole (BBH) parameter spaces [43], consisting of 1815963 templates in total. This bank is used for the GstLAL online analysis, as well as for the bulk of the GstLAL offline results. It extends from component masses of 1 $M_\odot$ to 200 $M_\odot$. The dimensionless spin components are restricted to be aligned with an orbital angular momentum vector to limit the dimensionality of the bank and their absolute values are capped at 0.99 for non-BNS templates, and 0.05 for BNS templates. The bank is an aligned-spin bank, to limit its size. Specifications of the AllSky template bank are shown in Tab. 4.1, and Fig. 4.1 shows a representation of the AllSky bank in the m_1- m_2 space. Readers are referred to [41] for more details about the AllSky template bank.

In comparison, the GstLAL O3 AllSky template bank covered component masses $m_i \in [1.0, 400]M_\odot$. It also limited templates to spin-aligned systems and set an upper limit of 0.05 on the magnitude of the spin for component masses below $3M_\odot$. The spin magnitudes for component masses above $3M_\odot$ were set to 0.999. The O3 template bank used a minimum match value of 99% for the BNS space, in contrast to the O4 value of 97%. The parameter choices of the O3 template bank is shown in Table II of [7]. As a result of different template bank generation algorithms and parameter choices, the O3 AllSky bank consisted of 1758763 templates.

Parameter	AllSky Template Bank
Primary mass, m_1	$\in [1.0, 200M_\odot]$
Secondary Mass, m_2	$\in [1.0, 200M_\odot]$
Mass ratio, $q = m_1/m_2$	$\in [1, 20]$
Total mass, $m_1 + m_2$	$\in [2.0, 400M_\odot]$
Dimensionless spin, $s_{\mathrm{i,z}}$, for $m_i \leq 3.0M_\odot$	$\lvert s_{\mathrm{i,z}}\rvert \leq 0.05$
Dimensionless spin, $s_{\mathrm{i,z}}$, for $m_i \geq 3.0M_\odot$	$\lvert s_{\mathrm{i,z}}\rvert \leq 0.99$
Lower frequency cut-off	10 Hz [1]
Higher frequency cut-off	512 Hz
Waveform approximant	IMRPhenomD
Minimum match	97 %
PSD	O4 projected PSD [2]
Total number of templates	1815963

Table 4.1. Parameters of the GstLAL O4 AllSky template bank.

4.2.1.2 IMBH template bank

In contrast to the AllSky bank, the IMBH bank is much smaller, and is only used to augment the GstLAL offline results in the IMBH parameter space. The IMBH bank covers the mass parameter space higher than what the AllSky template bank targets, as shown in Table 4.2. The lower limit of m_1 is set to 200 $M_\odot$ such that the IMBH bank is an extention to the AllSky bank and the templates of the two template banks do not overlap in the m_1-m_2 space, as shown in Fig. 4.1. The templates are limited to spin-aligned systems to reduce computational costs, similar to the AllSky bank. Details of the parameters of the IMBH bank are shown in Tab. 4.2. Fig. 4.1 also shows a plot of the IMBH bank in the m_1-m_2 space.

4.2.1.3 Population model

Population models provide weights to templates which represent our prior knowledge of the astrophysical distribution of GW sources [121]. This is used in the likelihood ratio (LR) calculation, as well as to compute probabilities of astrophysical origin of candidates [51]. In O4, the population models for both the AllSky and IMBH banks were generated using `manifold`. They implement a Salpeter mass function to assign weights to templates, defined as:

$$p(m_1, m_2, s_{1,z}, s_{2,z}) \propto \frac{m_1^{-2.35}}{m_1 - m_{\min}} \tag{4.1}$$

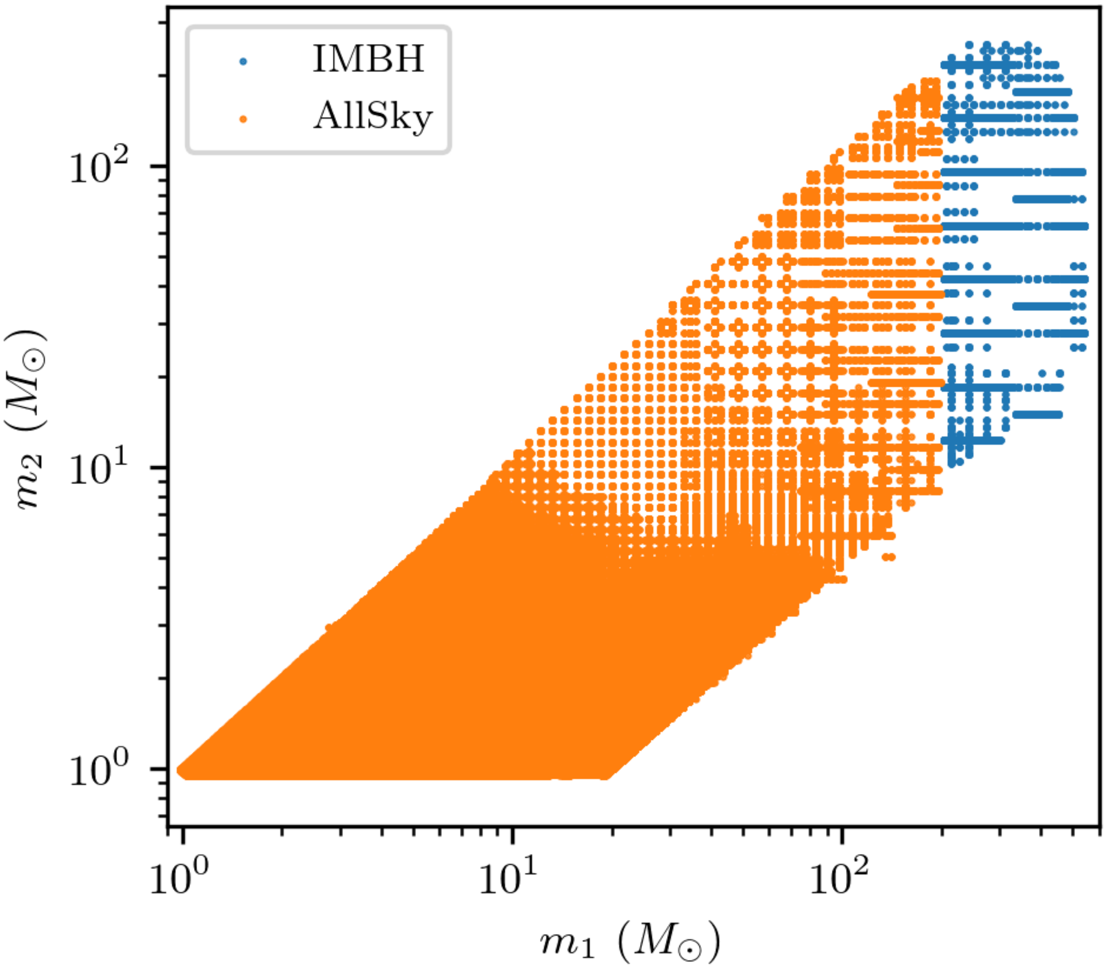


Figure 4.1. AllSky templates and IMBH templates on the log (m$_1$)-log (m$_2$) plane. Here, the orange dots with $m_1 \leq 200M_\odot$ are AllSky templates and blue dots with $m_1 \geq 200M_\odot$ are IMBH templates.

For both the AllSky and IMBH population models, the value of m_{min} is set to 0.8, which is slightly less than the minimum mass in the combined AllSky+IMBH template bank.

4.2.2 PSD measurement stage

The PSD is the frequency representation of detector noise, and is used in the matched filter calculation. This is done by whitening both the data and the templates with the PSD measured from the data. GstLAL has the ability to measure the PSD of the data in real time and dynamically whiten the data during the matched filtering stage. However, this ability does not extend to template whitening, and the templates need to be whitened

Parameter	IMBH Template Bank
Primary mass, m_1	$\in [200, 543 M_\odot]$
Secondary Mass, m_2	$\in [10, 258 M_\odot]$
Mass ratio, $q = m_1/m_2$	$\in [1, 20]$
Total mass, $m_1 + m_2$	$\in [50, 600 M_\odot]$
Dimensionless spin, $s_{\mathrm{i,z}}$, for m_i	$s_{\mathrm{i,z}} \in [-0.69, 0.98]$
Lower frequency cut-off	10 Hz
Higher frequency cut-off	512 Hz
Waveform approximant	IMRPhenomD
Minimum match	99 %
PSD	O4 projected PSD [3]
Total number of templates	14728

Table 4.2. Parameters of the GstLAL O4 IMBH template bank.

before the analysis starts. The PSD used for this purpose is measured from the data in the PSD measurement stage. The data is first divided into smaller pieces of up to 8 hours depending on the continuity of segments when each detector is operational. The PSD over each such piece of data is individually measured, and the median over each frequency bin is taken, which removes any transient noise features contributing to the PSD of a single piece of data. The resultant PSD is used to whiten the templates. More details about PSD measurement can be found in [19].

4.2.3 SVD of templates and template whitening stage

Next, as discussed in [41], the template bank is sorted by linear combinations of their Post-Newtonian phase coefficients and split into "template bins", each of around 1000 templates, which are whitened using the median PSD described above. These template bins further go through a process called SVD to save computational cost of the matched filtering [122]. The templates in one such template bin are processed together for the purpose of matched filtering and background estimation, constituting a single matched-filtering job, and hence this template binning allows for a large-scale parallelization in both online and offline workflows.

4.2.4 Matched filtering stage

Each template within a template bin is matched filtered with the data, producing triggers. Alongside SNR, a signal consistency statistic, called ξ^2 is also calculated for every trigger [19]. Triggers with contribution from only one detector (referred to as a "single detector trigger"), during a time when multiple detectors were operational (referred to as "coincident time"), are considered to originate from noise [42]. This is because GW signals are expected to be correlated across detectors, unlike noise. The SNR and ξ^2 statistics of noise triggers are added to the template bin's common noise background, which is later one of the inputs used to calculate the LR for triggers.

In parallel, the GW candidates are extracted from the triggers. This is done by retaining only those triggers within a template bin that have the highest SNR in a 0.2 second window. A single GW signal can create triggers via multiple templates, and with current detector sensitivities, it is highly unlikely for multiple GW signals to occur in the same 0.2 second window. Hence, this "clustering" over SNR, duplicate triggers from a single common signal, reducing the amount of data downstream stages need to process. The processes of populating the background and generating candidates are done in parallel for every template bin.

4.2.5 Rank stage

In this stage, the significance of candidates is evaluated, and the results of the search are produced. First, the candidate list is pruned again by performing a second round of clustering. This time, it is performed *across* template bins. The highest LR candidate with the highest LR in an 8 second window is retained. Apart from eliminating candidates originating from noise, this ensures that a single GW signal will have at most one candidate originating from it in the entire analysis.

The LR is then calculated for every surviving candidate. One of the ingredients in calculating the LR is the SNR-ξ^2 background populated by the matched filtering stage. To model a smooth distribution of SNR-ξ^2 statistics, a kernel density estimate (KDE) is applied to the background, and the resulting probability density is used for LR evaluation. This gives us the probability that the (SNR, ξ^2) of a given candidate arises from noise. This is one of the many terms in the LR equation. More details on the LR calculation can be found in [23].

Next, to convert the LR to a FAR, we need the LR distribution of noise triggers from every template bin [24, 25]. If we use the same set of noise triggers as was used to populate the SNR-ξ^2 background, the LR distribution will not be well defined at higher LRs due to lower number of statistics. Additionally, since these noise triggers were obtained during the livetime of the analysis, the lowest FAR would be bound by 1 per livetime. To solve this problem, GstLAL extrapolates the set of noise triggers from every template bin by drawing samples of SNR and ξ^2, and assigning random templates as well as arrival times and phases. This is a computationally cheap operation, and under the assumption that the template, time and phase distribution of noise triggers is uniform, this process can effectively extrapolate the LR distribution of noise triggers to high LRs, enabling the FAR calculation of even the most significant candidates.

However, this process does not work particularly well for low LRs. To fix this, we can make one more assumption, that at low LRs, even the candidates originate from noise. Consequently, the low LR distribution of candidates is the same as the low LR distribution of noise triggers, and the former can be used to inform the latter. The method of doing so is called the "extinction model". Once the LR distribution of noise triggers is calculated, it can be combined with the livetime of the analysis to produce a FAR for every candidate.

4.3 New Methods

4.3.1 Online Rank

During every observing run, the full data are analyzed in near-real time by the GstLAL low-latency or "online" analysis. The various stages of the online analysis, up to the matched filtering stage, are largely similar to those of the offline analysis. Consequently, during O4, a new method has been adopted to outsource the GstLAL offline analysis' matched filtering stage to the GstLAL online analysis. This method is called an "online rank". A description of the differences between the GstLAL online and offline analyses, as well as a detailed descreption of the online rank method is provided in [4].

As described in [4], the online rank method significantly reduces the computational cost and time needed for offline results. Matched filtering is the vast majority of the computational burden of a modeled GW search. By eliminating the need to repeat matched filtering a second time for the offline analysis, this method affords a 50% reduction in computational cost over the course of an observing run, and specifically a 95 - 99.8% reduction in time required to obtain offline results.

4.3.2 Dropped Data Refiltering

In order to make the online rank results even more reliable and sensitive, we can augment the inputs to the online rank with triggers and background data from times that the online analysis dropped. To do this, for every job in the online analysis, we take the list of time periods when each job was functional and producing output. We then subtract that from the list of times a traditional offline analysis would have analyzed, which is calculated by external tools as an extra step in the setup stage of the offline workflow. This leaves us with a list of times that a particular job has dropped. We calculate such a "dropped data segments" for every job. Then, we can set up a traditional offline analysis using these dropped data segments. The only unusual aspect of this procedure is that while setting up this "dropped data refiltering" analysis, each job has its own segments list rather than a global one for every job. An example of such a list of dropped data segments is shown in Fig. 4.2

By combining the online rank inputs with the results of the dropped data refiltering analysis, we can be sure the online rank produces offline results for exactly the same periods of time that the traditional offline analysis would have. The typical amount of dropped data for any job is around 5% of the total time covered by the offline segments [4].

4.3.3 IMBH Analysis

The AllSky template bank was designed to be a general-purpose bank for an online analysis. To obtain more comprehensive offline results, we want to augment the results of the AllSky search with a search for IMBH mergers. The GstLAL O4 offline results are obtained by performing a search over the data using the IMBH template bank described in Sec. 4.2, which is then combined with the AllSky search into a single set of results using the procedue described in the following subsection.

As shown in [22, 23], the LR contains a term to test the consistency of signals accross the network of detectors, for observed parameters like SNR and the coalescence time and phase. These parameters can only take on certain possible physical values for a network of detectors and follow specific correlations that indicate whether a GW trigger is of astrophysical origin. The correlation among these observables and their joint distribution depend on trigger's template, and hence in principle this signal consistency test should be modified for every template. However, for the purpose of pre-computation efficiency, we adopt a BNS template with component masses of 1.4 $M_{\odot}$, which is shown to be effective *throughout* Allsky template bank. In contrast, the mass range of the IMBH template bank is so high that this empirical fact might not be true, and hence for the IMBH analysis we use a template with component masses of 60 $M_{\odot}$.

Other differences in search settings between the AllSky and the IMBH searches include different lower frequency cutoffs during matched filtering. AllSky uses 15 Hz, but the IMBH search uses $10Hz$ as these binaries merge at lower frequencies. The Allsky

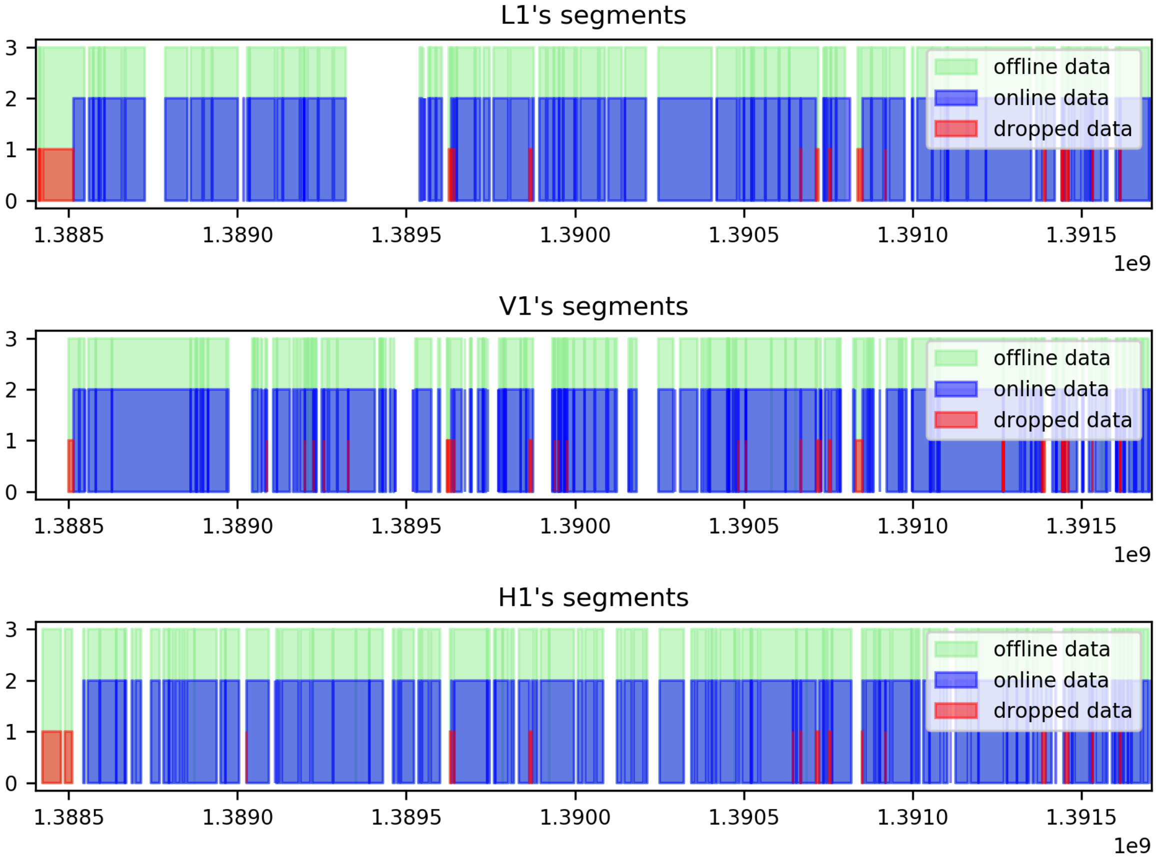


Figure 4.2. An example of a particular online job's list of dropped data segments. The dropped data segments at the very start of the plot are because of slightly different start times of the online and offline analysis, whereas the ones after that are because of online analysis having failed to analyze those times, either because the data for those times was dropped in order to keep up with incoming live data, or because the online job was not functioning at that time. We see that the amount of dropped data is not too large, but the segments are dispersed throughout the period of the analysis. [4] shows that a typical online analysis drops around 5% of the total data. This data can be filtered offline in order to augment the online rank's results. This particular plot was made using a GstLAL online analysis that participated in the mock data challenge [5].

search uses the TaylorF2 [123] approximant upto a chirp mass of 1.73 for cost effectivity and the SEOBNRv4_ROM [124] beyond that, while the IMBH search only uses the SEOBNRv4_ROM as an approximant. Similarly, the autocorrelation length used by the Allsky search for calculating the ξ signal-based veto [19] is 701 sample points upto a chirp mass of 1.73, and 351 above that. The IMBH search uses a value of 351 sample points throughout, due to a shorter duration of the waveforms. For this reason, the minimum number of samples included in a given time slice [19] is 512 for IMBH and 2048 for Allsky. For similar reasons, while splitting the IMBH bank into template bins, the templates are sorted by their duration which is found to be more effective for this part of the parameter space, rather than sorting by linear combinations of their Post-Newtonian phase coefficients.

It is known that the candidates in the IMBH search are overwhelmingly single-detector candidates. This was true even for triggers, before any clustering was done. For example, in the test described in Sec. 7.3, out of the total candidates reported by the IMBH search, 92.75% were found to be single-detector candidates, and out of the set of triggers, 98.57% were single-detector triggers. The corresponding numbers for the AllSky search are 3.37% and 35.07% respectively. While the root cause of this is not well understood, the vastly different percentages of single-detector triggers in the AllSky and IMBH searches indicate that this arises from the matched filtering and coincidence formation of IMBH templates, rather than anything to do with the LR.

The LRs of single-detector candidates are notoriously difficult to estimate [125], and so is the process of LR extrapolation of noise triggers. Consequently, the FARs of single-detector candidates are unreliable, and a search with such a high percentage of single detecotr candidates is unlikely to be functional. As a result, for GstLAL's O4 offline results, the IMBH search ignores single detector triggers, and only processes those with multiple contributing detectors. The effects on the LR statistics of candidates if single-detector triggers are allowed to be processed are shown in Fig. 4.3.

4.3.4 Combining the AllSky and IMBH analyses

In order to combine Allsky and IMBH searches into a single search, one can naively treat every template bin from both the AllSky and IMBH searches as though they are part of a bigger AllSky+IMBH joint template bank. This means after each template bin assigns LRs to its candidates, these candidates would be clustered together based on their LR, and the noise LR statistics of each template bin will be combined via the extinction model. An implicit assumption of this method is that each template bin's candidate and noise LR statistics in the noise regime are approximately the same. However, with the different parameter space and settings of the IMBH search as compared to AllSky, this assumption is no longer valid, and hence the naively combined results can be severely biased. To this end, we have developed a new scheme to combine the AllSky and IMBH analyses into a single search.

We first calculate a *weight* for each individual analysis, based on each analysis' contribution to the final clustered set of candidates. This was done using injection

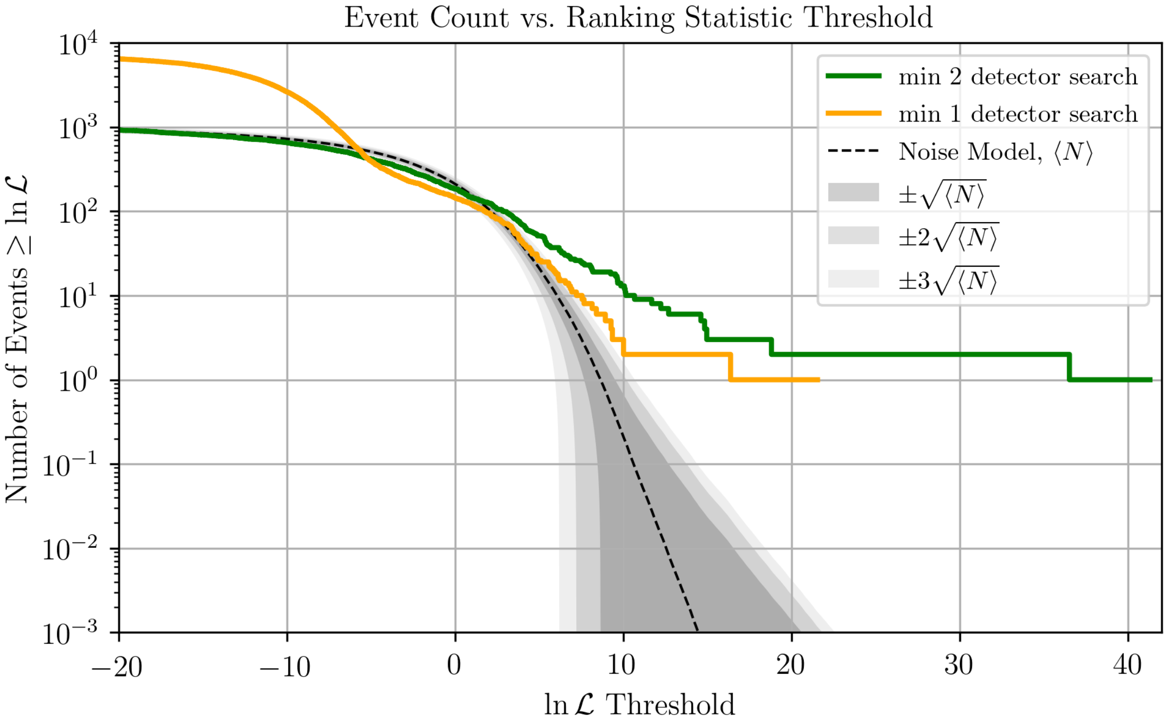


Figure 4.3. This plot shows the results of two IMBH searches: one which only considers triggers with two or more detectors contributing to it, and another with no such restriction. The plot also shows the noise LR statistics of the former search for reference. We can see that the LR statistics of the candidates of the search that only processes triggers with two or more detectors is well behaved, whereas the other one is not. This is because the IMBH search, if allowed to process single-detector triggers recovers an overwhelming amount of them, and the LRs (and hence FARs) of single-detector triggers are difficult to accurately calculate. As a result, for O4, GstLAL's IMBH search doesn't process single-detector triggers.

campaigns for both analyses. Injections are simulated GW signals inserted into the data. The weights for the AllSky and IMBH analyses was found to be 0.94 and 0.06. This result is also verified by summing over the population model weights for the AllSky and IMBH templates after taking into consideration the different SNR detection thresholds of the two analyses.

Next, the FARs of each analysis' candidates are scaled up by the inverse of their respective weights. Note that FARs are assigned to each analysis separately, and hence the FARs of candidates in the noise regime in both analyses are guaranteed to be distributed similarly as they only depend on the livetime of the search. Since the weights sum up to 1, this new set of candidates is guaranteed to have a consistent distribution of FARs in the noise regime, i.e., there will be on average livetime-per-hour number of candidates with a FAR of one per hour or lower, and so on. Candidates from both analyses are then clustered together. The LRs of the candidates are then re-calculated based on their new FARs, using the inverse of the mapping used by the AllSky analysis to convert LRs into FARs. In principle, any such mapping would have worked, and the AllSky mapping is chosen only for convenience.

4.3.5 Modularity and reusability of results

A general theme of the development work prior to and during O4 has been about reducing the duplication of computations done elsewhere. Apart from the online rank described in Sec. 4.3.1, we have made the GstLAL offline workflow more modular and flexible. In general, offline analyses over large periods of data (such as GstLAL's offline results for O4) are performed in small chunks, each of around 1 week of data. Alongside the ability to extract any amount of matched filtering data from the online analysis as specified by the user (to the granularity of 4 hours) [4], the GstLAL offline workflow also has the ability to use the data from any set of offline chunks specified by the user. Consequently, one can specify any period of data covered by any combination of online and offline analyses using any template banks, and they can be combined into a single result. Therefore, once a period of data has been matched filtered by a particular template bank via either an online (complemented by the dropped data refiltering) or offline analysis, it never needs to be filtered using that template bank again. This has greatly reduced the computational and time burden of GstLAL operations as well as development work during O4.

4.3.6 Ranking statistic improvements

GstLAL has adopted the likelihood ratio as the ranking statistic to evaluate the statistical significance of GW candidates [23, 24]. The likelihood ratio in GstLAL takes the form of

$$\mathcal{L} = \frac{P\left(\vec{O}, \vec{\rho}, \vec{\xi^2}, \vec{t}, \vec{\phi}, \theta \mid \mathcal{H}_\mathrm{s}\right)}{P\left(\vec{O}, \vec{\rho}, \vec{\xi^2}, \vec{t}, \vec{\phi}, \theta \mid \mathcal{H}_\mathrm{n}\right)}, \tag{4.2}$$

which represents the probability of obtaining a set of observable parameters, e.g., SNR for each detector $\vec{\rho}$, under the signal hypothesis ($\mathcal{H}_\mathrm{s}$) relative to that under the noise hypothesis ($\mathcal{H}_\mathrm{n}$). Here, ρ refers to the SNR of a single detector, and we will use this terminology for the rest of this subsection to maintain compatibility with others papers on this subject. The improvements in this ranking statistic prior to O4's start are thoroughly described in [23], which include the upgraded $\rho - \xi^2$ signal model and removal of signal contamination. In particular, the upgraded $\rho - \xi^2$ signal model achieved $\sim 20\%$ increase in the search sensitivity, and hence has led to a major contribution of GstLAL to LVK's GW detections in O4 so far.

Apart from the improvements mentioned above, we have modified the $\rho - \xi^2$ noise model to make more accurate estimate of event's significance in some part of (ρ, ξ^2) parameter space. As described in [23, 42], GstLAL collects single-detector triggers during coincident time into the 2D $\rho - \xi^2$ histogram and estimates its noise model. Since templates of lighter binary systems such as BNS tend to not couple with non-Gaussian noise, e.g., *glitches*, most of noise triggers associated with such low-mass templates are expected to be modeled by the Gaussian noise component. For each detector, trigger's SNR is calculated as square root of a quadrature sum of SNR values given by two orthogonal templates through matched-filtering, each of which follows the normal distribution. Therefore, the SNR-squared follows a chi-square distribution with two degree-of-freedom regardless of the observed ξ^2 value.

Also, according to the formalism of the upgraded $\rho - \xi^2$ signal model described in [23], the expected value of ξ^2 in the *noise* model can be also given by substituting the mismatch factor $k = 1$, i.e., maximum mismatch between a hypothetical signal and template, into Eq.(40) of [23],

$$\langle \xi^2 \rangle = \frac{1}{N} \left\{ N + (\rho^2 - 1) \vec{R}^\dagger \vec{R} \right\} \sim 1, \tag{4.3}$$

where N is a auto-correlation length used for ξ^2 calculation and $\vec{R}$ is an auto-correlation function of a given template normalized so that $R[0] = 1$. This approximation holds because an auto-correlation of compact binary coalescence (CBC) signals in general sharply peaks at the center where two templates exactly align, and hence $\vec{R}^\dagger \vec{R} \ll N(R[0])^2 = N$. Therefore, triggers from the Gaussian noise component are distributed near $\xi^2 = 1$ and decaying toward higher SNR. Consistently, this behavior is illustrated in Fig. 4.4, which shows a $\rho - \xi^2$ noise model for one of the low-mass template bins collected for LIGO Hanford detector during the mock data campaign described in [5] with the lightblue dashed line as a $\xi^2 = 1$ contour. In contrast, a tail of the distribution, which extends toward higher SNR and ξ^2, is known as non-Gaussian noise component.

However, during the online analysis of O4, we discovered that the Gaussian component of the noise model deviates from the expected chi-square distribution significantly at $\rho > 9$ because very few or zero noise trigger populates such a higher SNR regime and the KDE applied as a smoothing process does not complement the lack of triggers sufficiently.

See Fig. 4.5 where a 2D $\rho - \xi^2$ noise model is sliced at the $\xi^2 = 1$ contour and projected onto the ρ dimension. Note that the expected chi-squared distribution shown as a black curve is given by

$$p(\rho) = \chi_2^2(\rho)\frac{\mathrm{d}\rho^2}{\mathrm{d}\rho} \propto 2\rho \mathrm{e}^{-\rho^2/2}. \tag{4.4}$$

Although, this disagreement can eventually improve if we keep the analysis running and collect noise triggers long enough, early phase of the online analysis is likely to overestimate an event's significance due to this bias and send out a public alert, e.g., S240422ed [108][4]. Also, we cannot guarantee that this bias does not exist even in an offline analysis where noise triggers collected entirely from single (or subset of) observing run are considered. Therefore, the objective here is to modify the tail of the Gaussian noise component such that it can produce more accurate noise model even with fewer collected triggers and eventually prevent a potential false claim of GW detections.

To this end, after collecting enough noise triggers, we extrapolate the trigger counts along a given iso-ξ^2 contour using the bulk of the Gaussian component so that its tail part follows the expected distribution shown in Eq. (4.4). This process is iterated over a range of $-0.5 < \log \xi^2 < 0.5$ and finally the KDE is applied to produce a smoother distribution. The green curve in Fig. 4.5 shows that the noise model after this extrapolation closely follows the expected distribution in black curve. Although it can be slightly above the black curve, one can see that it still provides much more accurate estimate of the event's significance than the red curve without the extrapolation at $\rho > 9$.

4.3.7 New Extinction Model

A set of candidates undergo two rounds of clustering, the first with other candidates within the template bin, and the second with candidates from all template bins. These clustering processes alter the LR distribution of candidates. Consequently, the LR distribution of the noise triggers which gets used to convert the LR of candidates to FARs also needs to undergo these clustering processes. The new extinction model quantifies the change in LR distribution due to clustering, and applies it to the LR distribution on noise triggers. To mimic clustering, this is performed in two steps, the first within the noise triggers of a tempalte bin, and the second with noise triggers across template bins. Additionally, the new extinction model enforces that any template bin's contribution to

[4] The statistical significance of this candidate was subsequently estimated to be lower after collecting more noise triggers and the update alert was sent [126]

the noise LR distribution is proportional to its contribution to the LR distribution of candidates. Both of these features ensure that the FARs assigned to candidates are more accurate than the old method, i.e., stitchin the LR distribution of candidates below some LR threshold to the LR distribution of noise triggers [127].

Let $y(L)$ be the histogram of noise triggers that GstLAL stores in order to estimate the complementary cumulative distribution function of noise LRs, $N(L)$. The two are related in the following way:

$$N(L) = A\int_L^\infty y(L^*)dL^* \tag{4.5}$$

$$N_c(L) = A\int_L^\infty y_c(L^*)dL^* \tag{4.6}$$

where A is some normalization constant, and the subscript c denotes the clustered version of the respective functions. Equivalently,

$$y(L) = -\frac{dN(L)}{AdL} \tag{4.7}$$

The new extinction model assumes that the process of obtaining triggers above a certain LR threshold L^* in the clustering interval is a Poisson process, with Poisson rate given by:

$$\lambda = c \times N(L^*) \tag{4.8}$$

where c is another normalization constant. Hence the probability that some trigger with LR $= L^*$ survives clustering, i.e. that there are no triggers with a higher LR than L^* in the clustering interval is the Poisson probability for zero events.

$$P_{\text{survival}}(L^*) = e^{-\lambda} = e^{-cN(L^*)} \tag{4.9}$$

Hence, we can add the effect of clustering to $y(L)$, by multiplying it with the survival probability.

$$y_c(L) = y(L)e^{-cN(L)} \tag{4.10}$$

Substituting Eq. (4.10) in Eq. (4.6), it follows

$$\begin{aligned} N_c(L) &= A\int_L^\infty y(L^*)e^{-cN(L^*)}dL^* \\ &= -\int_L^\infty e^{-cN(L^*)}dN(L^*) \end{aligned}$$

$$= \frac{1}{c}[1 - e^{-cN(L)}]$$

where $N(\infty) = 0$. Since GstLAL deals with the complementary cumulative histogram of LRs, $n(L)$, rather than probability density, we can convert this to the histogram form by adding another normalization constant. After absorbing relevant constants in A and c, we get:

$$n_c(L) = A[1 - e^{-cn(L)}] \tag{4.11}$$

In order to find the constants A and c, the new extinction model performs a curve fit of $n_c(L)$ to the LR histogram of candidates. This is first done within a template bin, and after all of those clustered bin-specific noise histograms are added together, it is performed across template bins. The constant A takes care of the relative contributions of template bins to the candidate set, whereas the constant c takes care of the effect of clustering on the set of candidates. The curve fitting is performed from the 50th percentile to the 99th percentile of the complementary cumulative histogram of candidate LRs. This is empirically known to be a region of well-modeled noise candidates. The uncertainties in the curve fitting can be calculated from the assumption that a particular realization of the value of $y(L)$ is Poisson distributed. Hence,

$$\begin{aligned} \sigma_{y(L)} &= \sqrt{y(L)} \qquad (4.12) \\ \sigma^2_{N(L)} &= \int_{L^*}^{\infty} \sigma^2_{y(L^*)} dL^* \\ &= \int_{L^*}^{\infty} y(L^*) dL^* \\ \sigma_{N(L)} &= \sqrt{N(L)} \qquad (4.13) \end{aligned}$$

The effect of applying the new extinction model on the noise triggers is shown in Fig. 4.6. We can see that in the noise regime (i.e. 50th percentile to the 99th percentile), the noise LR histogram with the new extinction model is very close to the candiate histogram. This shows that the new extinction model is successful in applying the effects of candidate clustering and relative contributions of template bins to the candidate set. By adopting the new extinction model, we get more accurate FARs, since the noise LR distribution used to assign FARs is more accurate than the old method. The new extinction model has also been adopted for GstLAL's online operations in the second half of the fourth observing run (O4b).

4.4 Results

4.4.1 Data set

To test the new methods described in Sec. 7.2, and to serve as a trial run for GstLAL's O4 offline results, we set up a GstLAL analysis over 1 chunk of LIGO and Virgo O3 data. The data extends for approximately one week, from May 12 19:36:42 UTC 2019 to May 21 14:45:08 UTC 2019. Since [4] already tests the online rank feature and concludes it is equivalent to a traditional offline analysis, we do not test that feature here, and directly set up an offline analysis over the data instead of running an online analysis and setting up an online rank based on that. The offline analysis included a search with the AllSky bank, and a search with the IMBH bank, which were then combined into a single search using the procedure described in Sec. 7.2, referred to as "O4 analysis" hereafter. It also includes the LR improvements, the extrapolation in $\rho - \xi^2$ noise model and new extinction model. We then compared the results to GstLAL's O3 offline results over the same period of data, referred to as "O3 analysis" hereafter. Also, note that the O3 analysis uses the O3 template bank described in Sec. 4.2.1.1. Both analyses also included the same injection campaign. The details of the distribution of injections in this campaign can be found in [7, 128].

4.4.2 Candidate lists

This set of data contains 6 GW candidates previously reported in [7, 8]. Both the O3 and O4 analyses recover all 6 candidates in the list of top 10 candidates, as summarized in Tab. 4.3 and Tab. 4.4 respectively The O4 search recovers all 6 confidently with a FAR below 1/month, whereas the O3 search only recovers 5 of the 6 confidently.

4.4.3 Sensitivity comparison

To compare the sensitivities of the O4 and O3 analyses, we use the sensitive volume-time, or VT of the two analyses as a metric. It represents the volume of 4-dimentional spacetime where the search can typically identical signals. Since the times analyzed by both the O4 and O3 analyses is exactly the same, the VTs of the two analyses are a measure of their relative sensitivities. Furthermore, the VTs can be calculated for different mass ranges in order to estimate the relative sensitivities to different source classes. A plot of the ratio of the VT of the O4 analysis with and without the extrapolation in $\rho - \xi^2$

Rank	**FAR** (Hz)	**SNR**	**UTC Time**	**Found Detectors**	**Operating Detectors**	$\mathbf{m_1}$ $(M_\odot)$	$\mathbf{m_2}$ $(M_\odot)$	$\mathbf{s_{1z}}$	$\mathbf{s_{2z}}$
1	2.69×10^{-38}	24.51	2019-05-21 07:43:59	H1,L1	H1,L1	54.75	29.25	-0.2011	-0.2011
2	1.46×10^{-20}	13.35	2019-05-19 15:35:44	H1,L1	H1,L1	83.15	79.75	0.6033	0.6033
3	8.60×10^{-19}	14.14	2019-05-21 03:02:29	H1,L1	H1,L1,V1	213.1	137.4	0.8712	0.8712
4	3.72×10^{-16}	12.30	2019-05-13 20:54:28	H1,L1,V1	H1,L1,V1	50.36	48.29	0.6033	0.6033
5	3.56×10^{-12}	10.48	2019-05-17 05:51:01	H1,L1	H1,L1,V1	50.36	40.86	0.7889	0.7889
6	1.65×10^{-10}	10.88	2019-05-20 10:35:19	L1,V1	H1,L1,V1	442.1	218.2	0.8646	0.8646
7	9.71×10^{-9}	8.41	2019-05-14 06:54:16	H1,L1	H1,L1	54.75	57.08	-0.2320	-0.2320
8	1.25×10^{-8}	13.00	2019-05-15 06:27:00	H1,L1	H1,L1	543.3	63.58	0.2638	0.2638
9	1.46×10^{-8}	13.67	2019-05-14 04:01:29	H1,L1	H1,L1	414.9	144.7	0.8778	0.8778
10	1.88×10^{-8}	8.39	2019-05-18 20:43:46	L1,V1	H1,L1,V1	414.9	218.2	0.8646	0.8646

Table 4.3. The candidate list for the offline search using GstLAL's O4 methods. The search is run over a week of O3 data. The first, second, third, fourth, fifth, and seventh candidates in this list correspond to the GWs GW190521_074359, GW190519_153544, GW190521, GW190513_205428, GW190517_055101, and GW190514_065416, previously reported in [7,8]. The remaining 4 candidates all arise from the IMBH search. Given the uncertainty surrounding the IMBH space of GW signals, we make no claims regarding the origin of these 4 candidates

Rank	**FAR** (Hz)	**SNR**	**UTC Time**	**Found Detectors**	**Operating Detectors**	$\mathbf{m_1}$ $(M_\odot)$	$\mathbf{m_2}$ $(M_\odot)$	$\mathbf{s_{1z}}$	$\mathbf{s_{2z}}$
1	1.04×10^{-43}	24.43	2019-05-21 07:43:59	H1,L1	H1,L1	56.33	37.81	-0.0015	0.4993
2	4.33×10^{-21}	11.68	2019-05-13 20:54:28	H1,L1	H1,L1	60.74	32.32	0.8530	-0.1904
3	1.58×10^{-18}	13.03	2019-05-19 15:35:44	H1,L1	H1,L1	103.40	11.61	0.0925	-0.9660
4	2.35×10^{-11}	10.36	2019-05-17 05:51:01	H1,L1,V1	H1,L1,V1	49.39	39.63	0.6712	0.8675
5	2.80×10^{-10}	12.94	2019-05-21 03:02:29	H1,L1,V1	H1,L1,V1	139.70	2.92	-0.2471	-0.0434
6	3.23×10^{-9}	8.42	2019-05-14 23:48:57	L1,V1	L1,V1	2.34	1.24	-0.0130	0.0336
7	3.16×10^{-7}	8.93	2019-05-20 06:39:01	H1,L1	H1,L1	4.94	1.10	0.3486	-0.0499
8	2.70×10^{-6}	8.33	2019-05-14 06:54:16	H1,L1	H1,L1	53.48	53.48	-0.5578	-0.5578
9	3.02×10^{-6}	9.63	2019-05-20 20:45:36	H1,V1	H1,L1,V1	21.42	18.44	0.6018	0.4931
10	3.85×10^{-6}	8.92	2019-05-17 14:43:16	H1,L1	H1,L1	19.96	1.24	-0.9457	0.0449

Table 4.4. The candidate list for the offline search using GstLAL's O3 methods. The search is run over a week of O3 data. The first, second, third, fourth, fifth, and eighth candidates in this list correspond to the GWs GW190521_074359, GW190513_205428, GW190519_153544, GW190517_055101, GW190521, and GW190514_065416, previously reported in [7,8].

noise model, with the O3 analysis is shown in Fig. 4.7.

4.4.4 Performance of individual features

4.4.4.1 IMBH analysis

Here, we assess how much sensitivity we gain by combining the IMBH search with the AllSky search, compared to just the sensitivity of the AllSky search. The VT ratio of the O4 combined AllSky+IMBH search to that of the VT of the O4 AllSky search is shown in Fig. 4.8. In addition to the 6% - 7% sensitivity improvement in the IMBH space as shown in Fig. 4.8, by adding the IMBH search to the AllSky one, the combined analyiss becomes sensitive to a new parameter space, and has the ability to recover GW signals arising from intermediate-mass black hole mergers, which the AllSky search did not have by itself. This source class is particularly interesting and by detecting GWs from such mergers, we enable new scientific results [129].

4.4.4.2 Extrapolation in $\rho - \xi^2$ noise model

As mentioned previously, the extrapolation in $\rho - \xi^2$ noise model removes false positives in GW candidates. This implies that real GW signals can also be downranked accordingly and not be recovered as significantly as it would be without the extrapolation. To this end, we compare the VTs between with and without the extrapolation to assess the signal recovery performance. Fig. 4.9 shows the VT of the O4 analysis with the noise model extrapolation as a function of the recovered FAR for each category of injections, being relative to that for without the extrapolation. Note that the sensitivity can decrease up to 15% for two lowest mass categories of the injections, e.g., blue and orange curves. This is because, as mentioned above, the noise model for lower mass templates tends to be dominated by the Gaussian component and hence the extrapolation is likely to be more impactful. In contrast, for heavier BBH templates, the non-Gaussian noise component overwhelms the Gaussian one and makes small changes at its tail part negligible. We also emphasize that this decrease in the search sensitivity does *not* necessarily indicate the lowered performance overall, but rather that the signal recovery without the extrapolation is overestimated and risks potential false positives.

4.4.4.3 Reliability of results

The new extinction model and LR improvements ensure the FARs of candidates are more accurate, and the extrapolation in $\rho - \xi^2$ noise model removes potential false positives in the candidate set. All of these contribute to the O4 results being more reliable then before. This effect is difficult to measure since wrongly assigned FARs and false positives are rare. One possible metric is to see how many candidates in the injection campaign with a FAR below some threshold do not correspond to an injected GW signal. With a FAR threshold of one per day, this number for the O3 search is 53, representing 0.07% of all candidates. The corresponding number for the O4 search is 28, representing 0.03% of all candidates. For a higher FAR threshold of one per hour. the numbers for the O3 and O4 searches is 276 (0.35%) and 132 (0.15%). This shows that the new O4 methods are successful in lowering the number of false positives.

The O4 search includes the IMBH search. This parameter space is known to be particularly vulnerable to loud noise transients or non-gaussian transient "glitches". If we only look at the O4 AllSky search, for a more direct comparison with O3, the numbers for the one-per-day and one-per-hour FAR thresholds are 23 (0.03%), and 110 (0.13%), showing an even higher efficacy of removing false positives.

4.5 Conclusion

In this work, we first gave an overview of the GstLAL offline analysis. This included desriptions of the GstLAL AllSky and IMBH template banks used in O4. We later described new methods in the GstLAL offline analysis introduced in the lead up to and during O4. These new methods are all used to obtain GstLAL's offline results for O4.

The new methods include a way to outsource everything up to the matched filtering stage to an online analysis that had previously run on the same data. This, along with improvements to modularity of the workflow and reusability of results means that GW data only needs to be matched filtered with a given template bank only once during an observing run. The results of any such matched filtering, whether online or offline, and with any template bank can be combined into a single cohesive set of results.

The new methods also include a new IMBH offline search. This search only allows candidates with multiple contributing detectors, in order to make the search well-behaved and functional. A new method to combine the AllSky and IMBH searches was also introduced. It assigns a weight to each search by which the FARs of their candidates, calculated independently for each search, are scaled up. Afterwards, the set of candidates of the two searches can be combined in to single set of candidates.

Finally, we also discussed some changes to the LR calculation, introduced the extrapolation in $\rho - \xi^2$ noise model, as well as a new extinction model which enables a better estimation of the LR statistics of noise triggers.

To test these new methods, we set up an offline analysis over one week of O3 data, and compared the results with those obtained from the same data by GstLAL during O3. Both searches recovered the 6 previously reported GWs in the data in their list of top 10 candidates. The O4 search was able to do so confidently for all 6, whereas the O3 search managed it only for 5. We found that the sensitivity of the O4 search in the IMBH space increased by 50% - 100%. The VT in the BNS and NSBH space went down by 10% - 20% as a result of the extrapolation in $\rho - \xi^2$ noise model, which reduces false positives and downweighting a small number of real GW signals accordingly. Without the new KDE, those VTs are very close to the O3 ones.

The new methods described in this work variously increase the sensitivity, reliability, or reusability of the GstLAL offline results. As a result, we expect GstLAL's offline results to significantly contribute to the LVK's scientific results in O4.

This research has made use of data, software and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org/), a service of LIGO Laboratory, the LIGO Scientific Collaboration (LSC) and the Virgo Collaboration. We especially made heavy use of the LVK Algorithm Library. LIGO was constructed by the California Institute of Technology and the Massachusetts Institute of Technology with funding from the United States National Science Foundation (NSF) and operates under cooperative agreements PHYS-0757058 and PHY-0823459. In addition, the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany supported the construction of Advanced Laser Interferometer Gravitational-Wave Observatory (aLIGO) and construction and operation of the GEO600 detector. Additional support for aLIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain.

The authors are grateful for computational resources provided by the the LIGO Lab culster at the LIGO Laboratory and supported by PHY-0757058 and PHY−0823459, the Pennsylvania State University's Institute for Computational and Data Sciences gravitational-wave cluster, and supported by OAC-2103662, PHY-2308881, PHY-2011865, OAC-2201445, OAC-2018299, and PHY-2207728. The authors thank Reed Essick (and the entire LVK Rates and Populations group) for developing the injection sets used in the calculation of the weights for the AllSky and IMBH analyses. LT acknowledges support from the Nevada Center for Astrophysics. CH Acknowledges generous support from the Eberly College of Science, the Department of Physics, the Institute for Gravitation and the Cosmos, the Institute for Computational and Data Sciences, and the Freed Early Career Professorship.

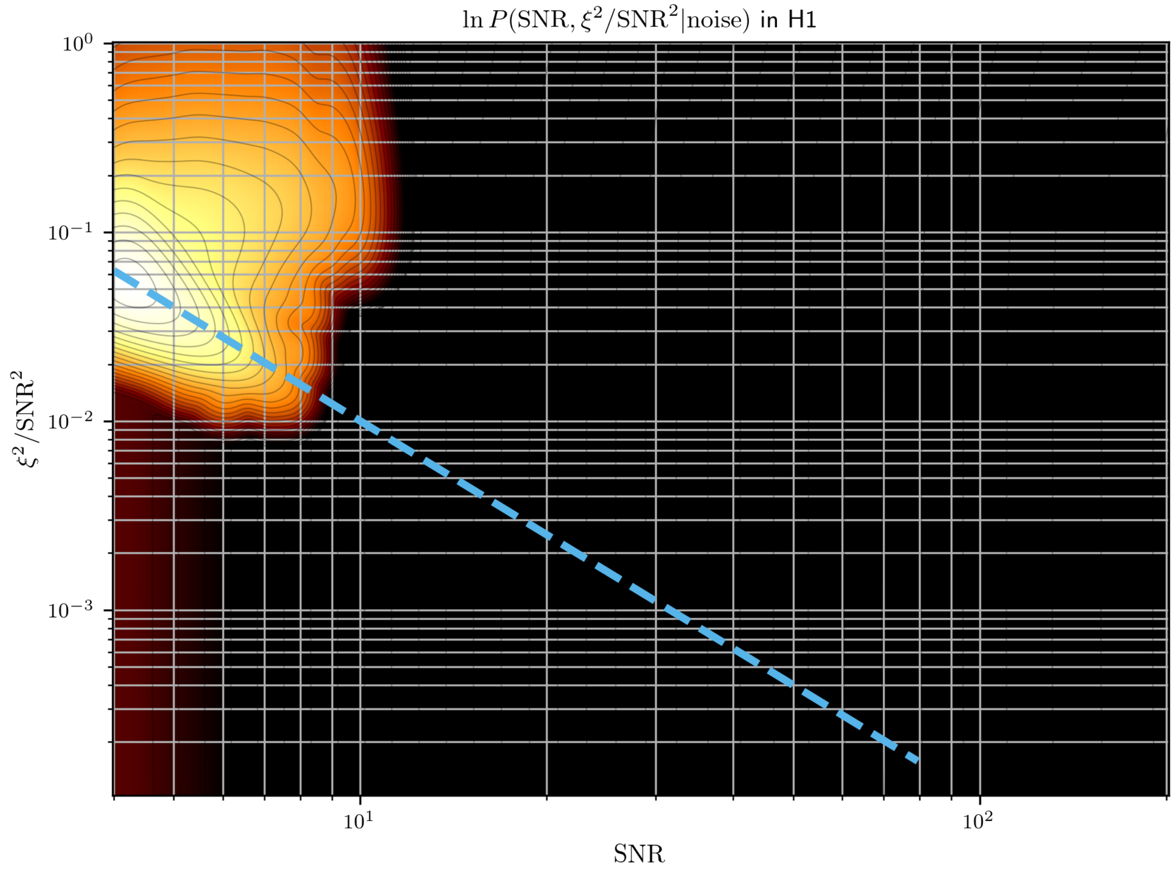


Figure 4.4. Example of $\rho - \xi^2$ noise model for one of the low-mass template groups collected for LIGO Hanford detector during the mock data campaign described in [5] with the lightblue dashed line as a $\xi^2 = 1$ contour

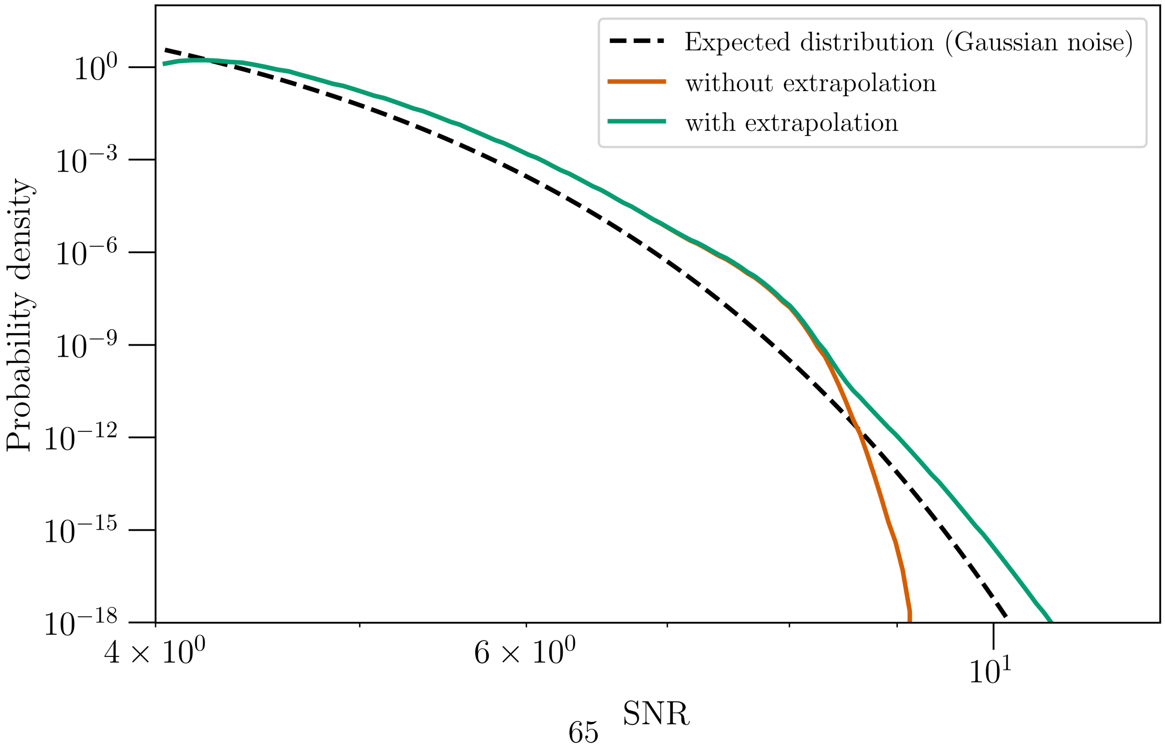


Figure 4.5. 2D $\rho - \xi^2$ noise model sliced at the $\xi^2 = 1$ contour and projected onto the ρ dimension. The noise model with the extrapolation closely follows the expected distribution in black curve at $\rho > 9$ as opposed to the red curve without the extrapolation.

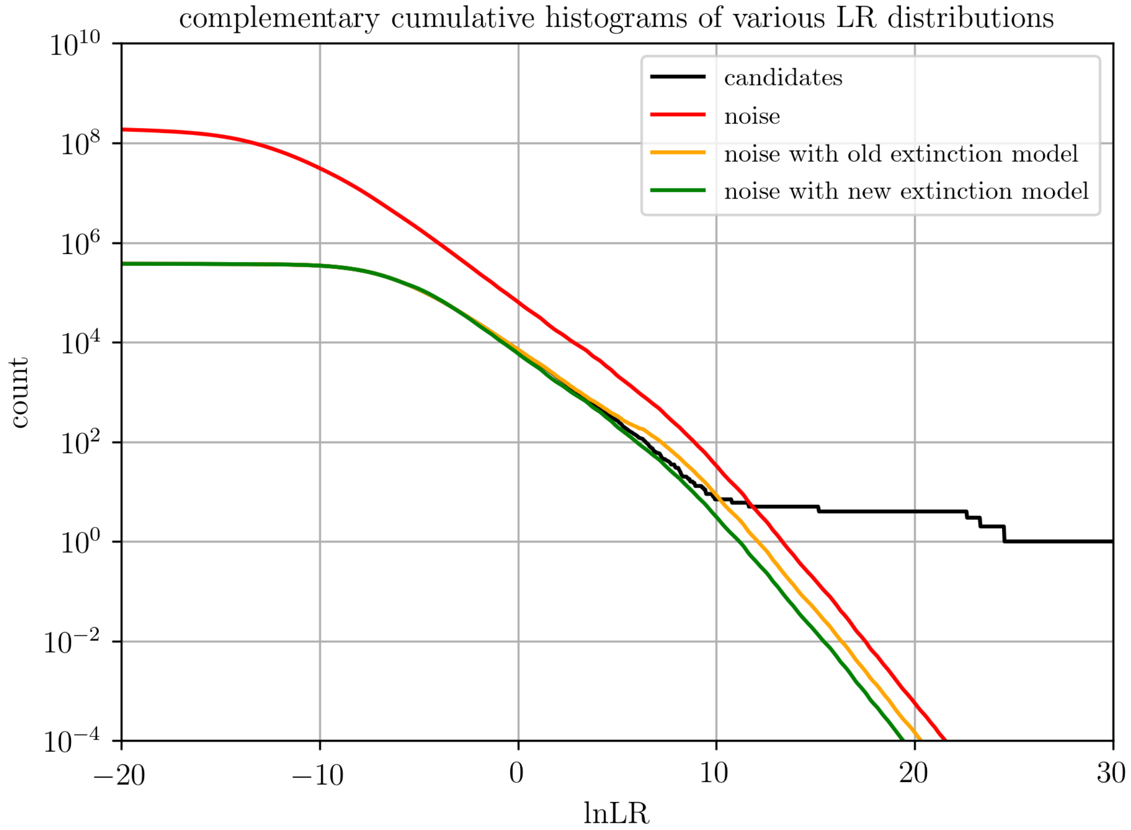


Figure 4.6. Effect of applying the new extinction model on the noise LR histogram. The effects of candidate clustering and differing relative contributions of template bins to the candidate set are modeled by modifying the noise LR histogram, n(L) to $A[1 - e^{-cn(L)}]$. The values of A and c are found by curve fitting the modified noise LR histogram to the candidate LR histogram. We can see that this process is effective from the fact that the noise LR histogram with the new extinction extinction is very close to the candidate LR histogram in the noise region (low LR).

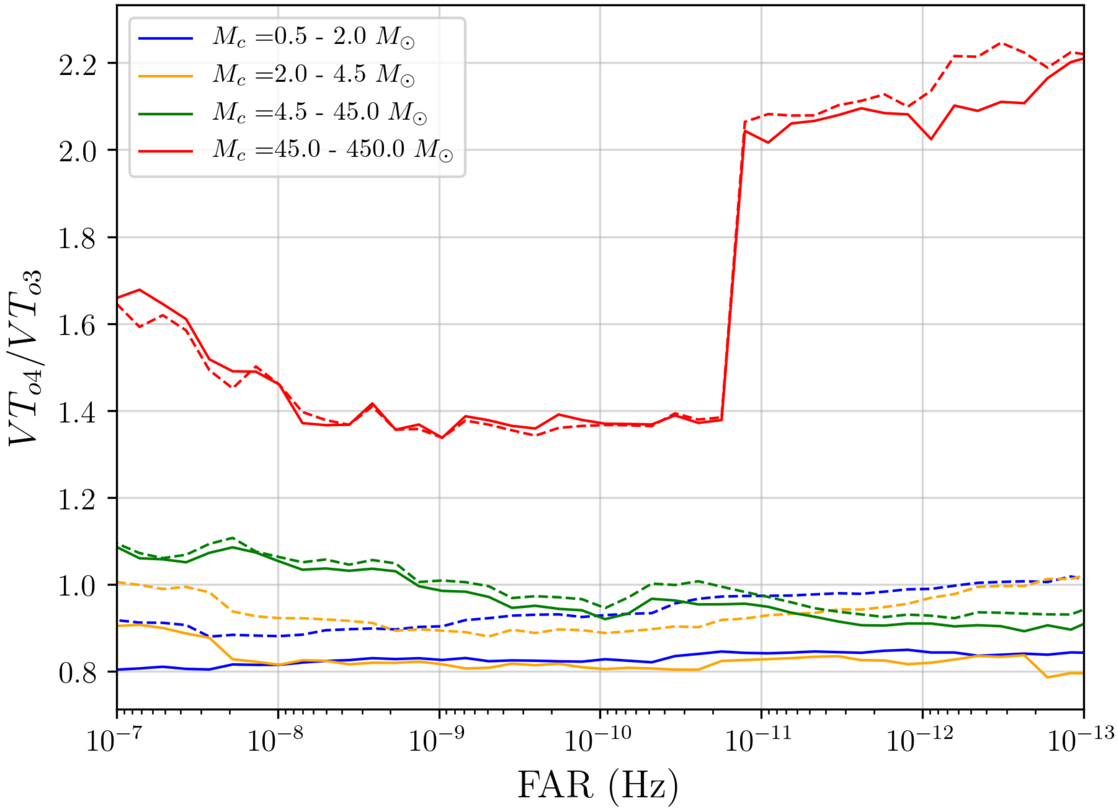


Figure 4.7. VT ratios of the O4 search with the O3 search. The solid lines represent the full O4 search as compared to the O3 search, whereas the dashed lines represent the O4 search without the extrapolation in $\rho - \xi^2$ noise model as compared to the O3 search. Since the extrapolation removes false positives from the candidates, it corrects previous mis-estimations of the VT done in the O3 search, superficially lowering the O4 VT. The VT ratios of the two lowest mass bins are slightly less than 1. This is because the O3 temlpate bank sampled the lower mass parameter space with a higher minimum match (0.99) as compared to O4 (0.97). Additionally, we see that the highest mass bin has a 50% - 100% increase in sensitivity, arising from the LR improvements and the additional IMBH search done in O4.

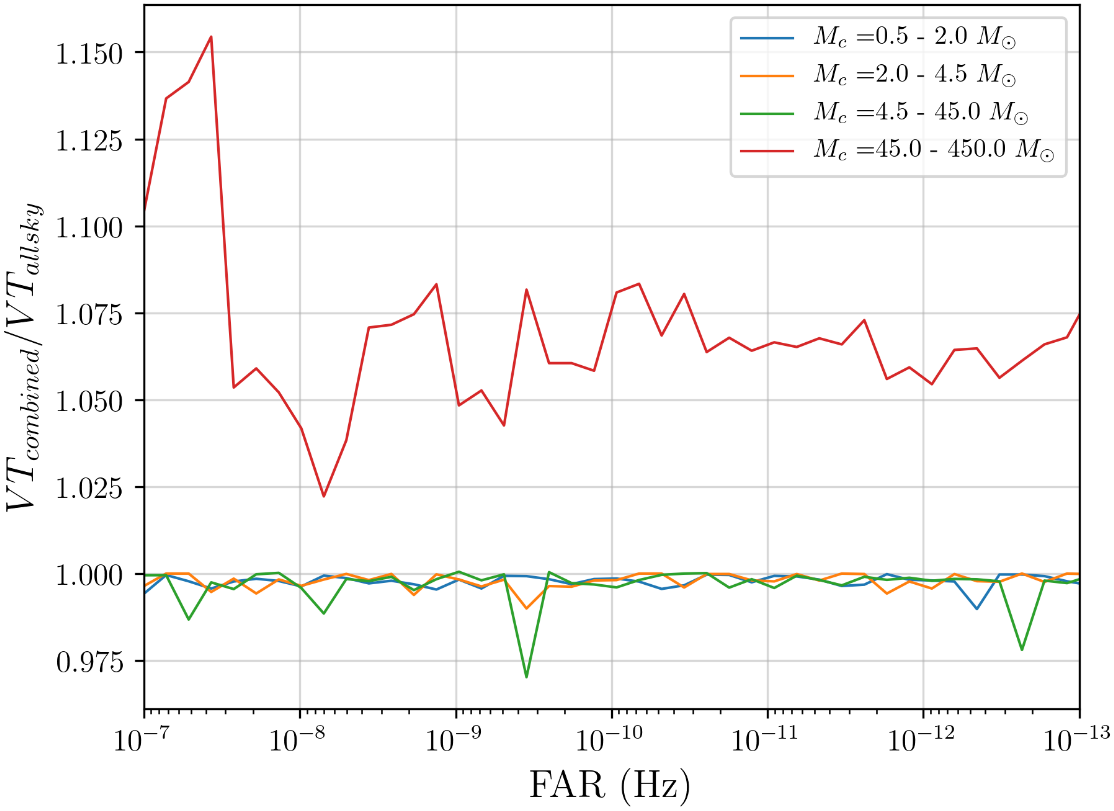


Figure 4.8. This plot shows the ratio of VT of the O4 combined AllSky+IMBH search to that of the O4 AllSky search, and it shows us how much sensitivity we gain by combining the IMBH and AllSky searches, as compared to just the AllSky search. As expected the sensitivities of the 3 lowest mass bins are unchanged, since the IMBH bank is not sensitive in that region. The VT of the IMBH bin increases by 6% - 7%.

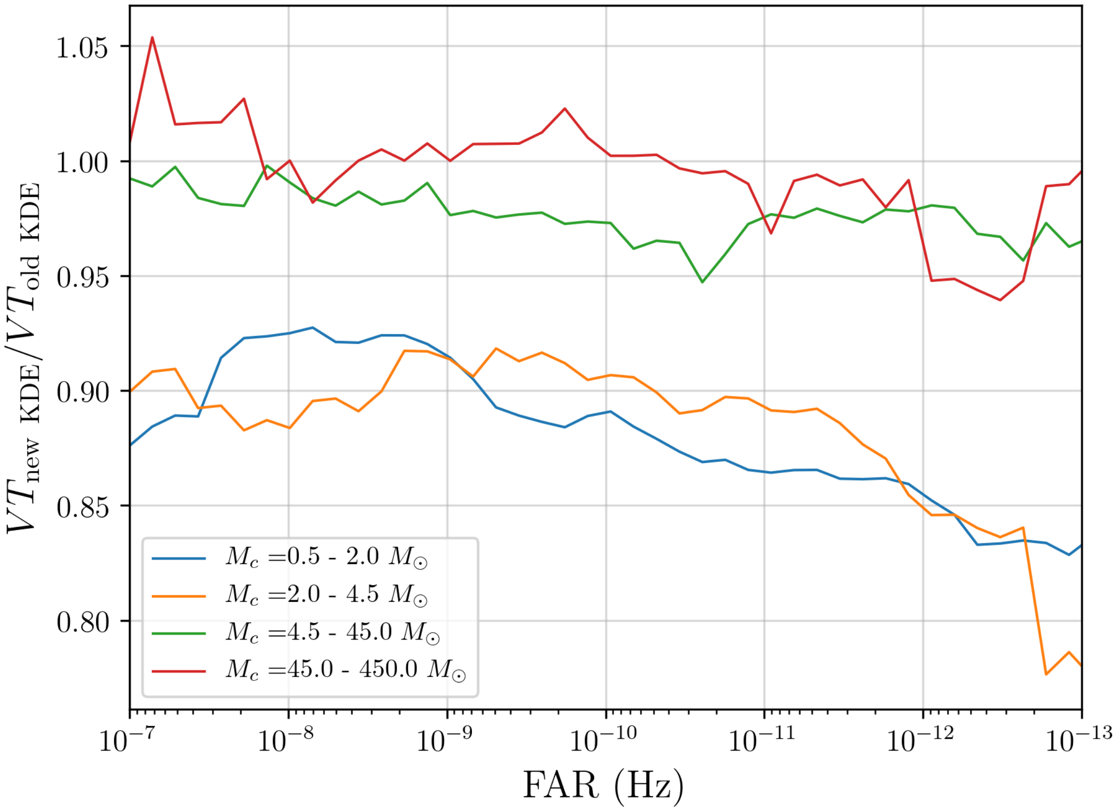


Figure 4.9. VT ratio of the O4 search with the noise model extrapolation as compared to without. The extrapolation removes false positives from the set of candidates, and in the process also downweights a small amount of real GW signals. The former effect corrects the overestimate of VT that happens without the extrapolation, whereas the latter decreases the real sensitivity of the search. Both these effects contribute to the lower than 1 VT ratio seen here. This effect is mostly seen in the BNS ans NSBH regions, whose VT goes down by 10% - 15%. The BBH and IMBH regions are mostly unaffected.

Chapter 5
Method for removing signal contamination during significance estimation of a GstLAL anaysis

This chapter is a reprint of [42]

To evaluate the probability of a gravitational-wave candidate originating from noise, GstLAL collects noise statistics from the data it analyzes. Gravitational-wave signals of astrophysical origin get added to the noise statistics, harming the sensitivity of the search. We present the Background Filter, a novel tool to prevent this by removing noise statistics that were collected from gravitational-wave candidates. To demonstrate its efficacy, we analyze one week of LIGO and Virgo O3 data, and show that it improves the sensitivity of the analysis by 20-40% in the high mass region, in the presence of 868 simulated gravitational-wave signals. With the upcoming fourth observing run of LIGO, Virgo, and KAGRA expected to yield a high rate of gravitational-wave detections, we expect the Background Filter to be an important tool for increasing the sensitivity of a GstLAL analysis.

5.1 Introduction

The Laser Interferometer Gravitational-wave Observatory (LIGO) [11] and Virgo [12] collaborations have revolutionized the field of gravitational-wave (GW) astronomy by detecting black hole and neutron star mergers [6–8, 14]. The detections have allowed us to observe the universe in new ways and have opened up new avenues of scientific inquiry. [57, 60, 62, 130] The GstLAL GW search pipeline [19–22] (referred to as GstLAL hereafter) has been a significant contributor to this field. In particular, GstLAL's ability to detect signals in low-latency [36] has facilitated multi-messenger observations [131].

GstLAL is a GW search pipeline that can process data from ground-based GW detectors, such as the Hanford and Livingston LIGO detectors, the Virgo detector and the KAGRA detector [13], in near real time. It makes use of time-domain matched-filtering to enable the detection of signals in noise-dominated data. It uses a likelihood ratio (LR) [23–25] as a ranking statistic for assigning significance to detections. GstLAL divides its template bank [41, 132] into different "template bins" to reduce the computational cost of the analysis, and analyzes each one separately. Some of these techniques are also used by other search pipelines, such as PyCBC [26–28], MBTA [29, 30], SPIIR [31, 32], and IAS [33, 34].

The fourth observing run of the LIGO Scientific, Virgo and KAGRA collaboration (O4) is set to begin in May 2023 [133] and promises to provide improved detector sensitivity. GstLAL will continue to play an essential role in the detection of new GW candidates. As such, it is necessary to keep refining the analysis pipeline to reap the benefits of improved detector sensitivity to detect even more, and new types of candidates. The Background Filter is one such new feature to this end.

This paper is structured as follows. In Sec. 5.2, we introduce the LR used by GstLAL, in particular the $\rho-\xi^2$ histograms that GstLAL uses to evaluate one term of the likelihood ratio, and how the presence of GW signals in the data can cause "contamination" of these histograms. In Sec. 5.3, we describe how the Background Filter works, and how it removes this contamination. Finally, in Sec. 7.3, we describe the analyses we performed to evaluate the performance of the Background Filter, and the impact it has on the sensitivity of a GstLAL analysis.

5.2 Signal Contamination

5.2.1 Likelihood Ratio

GstLAL is a matched-filtering based GW search pipeline which uses a likelihood ratio statistic to rank GW candidates [23, 24]. The LR is defined as

$$\mathcal{L} = \frac{P\left(\vec{O}, \vec{\rho}, \vec{\xi^2}, \vec{t}, \vec{\phi}, \bar{\theta} \mid \mathcal{H}_\mathrm{s}\right)}{P\left(\vec{O}, \vec{\rho}, \vec{\xi^2}, \vec{t}, \vec{\phi}, \bar{\theta} \mid \mathcal{H}_\mathrm{n}\right)}, \tag{5.1}$$

where the numerator is the probability of obtaining a GW candidate with parameters $(\vec{O}, \vec{\rho}, \vec{\xi^2}, \vec{t}, \vec{\phi}, \bar{\theta})$, under the signal hypothesis $(\mathcal{H}_\mathrm{s})$ and the denominator is the probability of obtaining the same candidate under the noise hypothesis ($\mathcal{H}_\mathrm{n}$). $\vec{O}$ is the subset of GW detectors that the candidate was found in, $\vec{\rho}$ is the set of matched-filtering signal-to-noise-ratios (SNRs) of those detectors, $\vec{\xi^2}$ is the set of ξ^2-signal-based-veto values, $\vec{t}, \vec{\phi}$ are the times and phases with which the candidate was found in the detectors, and $\bar{\theta}$ is the template which recovered the candidate, which also represents a set of intrinsic parameters (masses and spins).

The LR can be factorized as

$$\begin{aligned}\mathcal{L} = &\frac{P\left(\bar{\theta} \mid \mathcal{H}_\mathrm{s}\right) \times P\left(t_\mathrm{ref}, \phi_\mathrm{ref} \mid \bar{\theta}, \mathcal{H}_\mathrm{s}\right) \times P\left(\vec{O} \mid t_\mathrm{ref}, \mathcal{H}_\mathrm{s}\right)}{P\left(t_\mathrm{ref}, \bar{\theta} \mid \mathcal{H}_\mathrm{n}\right) \times P\left(\vec{O} \mid t_\mathrm{ref}, \bar{\theta}, \mathcal{H}_\mathrm{n}\right)} \\ &\times \frac{P\left(\vec{\rho}, \vec{\Delta t}, \vec{\Delta \phi} \mid \vec{O}, t_\mathrm{ref}, \mathcal{H}_\mathrm{s}\right) \times P\left(\vec{\xi^2} \mid \vec{\rho}, \bar{\theta}, \mathcal{H}_\mathrm{s}\right)}{P\left(\vec{\Delta t}, \vec{\phi} \mid \vec{O}, \mathcal{H}_\mathrm{n}\right) \times P\left(\vec{\rho}, \vec{\xi^2} \mid t_\mathrm{ref}, \bar{\theta}, \mathcal{H}_\mathrm{n}\right)}\end{aligned} \tag{5.2}$$

For a comprehensive explanation of Eq. (5.2) and every individual term in the LR, readers are referred to [23]. For the purpose of this paper, we are only concerned with the last term in the denominator, $P\left(\vec{\rho}, \vec{\xi^2} \mid t_\mathrm{ref}, \bar{\theta}, \mathcal{H}_\mathrm{n}\right)$ (hereby referred to as the $\rho - \xi^2$ noise LR term).

5.2.2 The $\rho - \xi^2$ histograms

The $\rho - \xi^2$ noise LR term is calculated in a data-driven way. GstLAL creates a histogram for each detector and template bin in $\rho - \xi^2$ space, called $\rho - \xi^2$ background histograms, and populates it with the (ρ, ξ^2) values of noise events found in that template bin during the analysis. Then, the $\rho - \xi^2$ noise LR term can be calculated by evaluating the probability density function represented by the histograms at the relevant ($\vec{\rho}$, $\vec{\xi^2}$) value.

Since the $\rho - \xi^2$ noise LR term assumes the noise hypothesis, we need to populate the histograms with events originating only from noise, as compared to events originating from GW candidates. To a large degree, this is achieved by requiring those events to be recovered only in one detector (called a single-detector or single event in contrast to a coincident event) during a time when more than one detector was producing data (called coincident time in contrast to single time). This is because we expect GW signals to be correlated across detectors, but not noise events.

Despite this, GW signals can sometimes enter the $\rho - \xi^2$ histograms. The reason might be astrophysical in origin, e.g. the GW source is located in the blind spot of all but one detector, or it might be terrestrial, e.g. only one detector is sensitive enough to pick up the GW signal. In addition, GW signals which are recovered as coincident events in one template bin sometimes also get recovered as a single event, with a lower ρ and LR in other neighbouring template bins, which don't contain templates with high ρ for that GW signal. We say a bin has a good match with a GW signal if it has templates with a high ρ for that GW signal, and that it has a bad match otherwise. GW events being recovered as coincident events in one bin and as single events in others is demonstrated in Fig. 5.1 for GW200129_065458, a known GW candidate reported in GWTC-3 [6]. The candidate is recovered as a coincident event in bin 818, with which it has the best match. It is also recovered in bin 838 as a Livingston single with a lower ρ, since it's match with that bin isn't as good. As a result, the candidate will be added to the background histogram of bin 838. Gravitational wave signals entering the background histograms is commonly called signal contamination of the $\rho - \xi^2$ background histograms. The contamination caused by GW200129_065458 in the background histogram of bin 838 is shown in Fig. 5.2. Since the GW signal gets added to the background histogram, it occupies a region in $\rho - \xi^2$ space typical of signals, but not of noise. As a result, we see a protrusion to the histogram, which is generally how signal contamination manifests visually.

Signal contamination can result in the $\rho - \xi^2$ histograms not accurately reflecting the noise characteristics of the data, and as a result, the $\rho - \xi^2$ noise LR term will not be calculated correctly. In general, it can cause the $\rho - \xi^2$ noise LR term for GW candidates to be evaluated higher than it's true value, leading to lower LR values of candidates. In short, signal contamination can lower the sensitivity of the GW search.

5.3 Removing contamination with the Background Filter

To prevent any loss in sensitivity due to signal contamination, we need to selectively remove the events in the $\rho - \xi^2$ background histograms which originate from GW signals. The Background Filter is a way to track the background in a time-dependent fashion so that we only use events from times not corresponding to GW events to populate the background histograms. In this paper, we will describe the working of the Background Filter when GstLAL is running in the low-latency online mode, in which data is analyzed and results are produced in near real time [43].

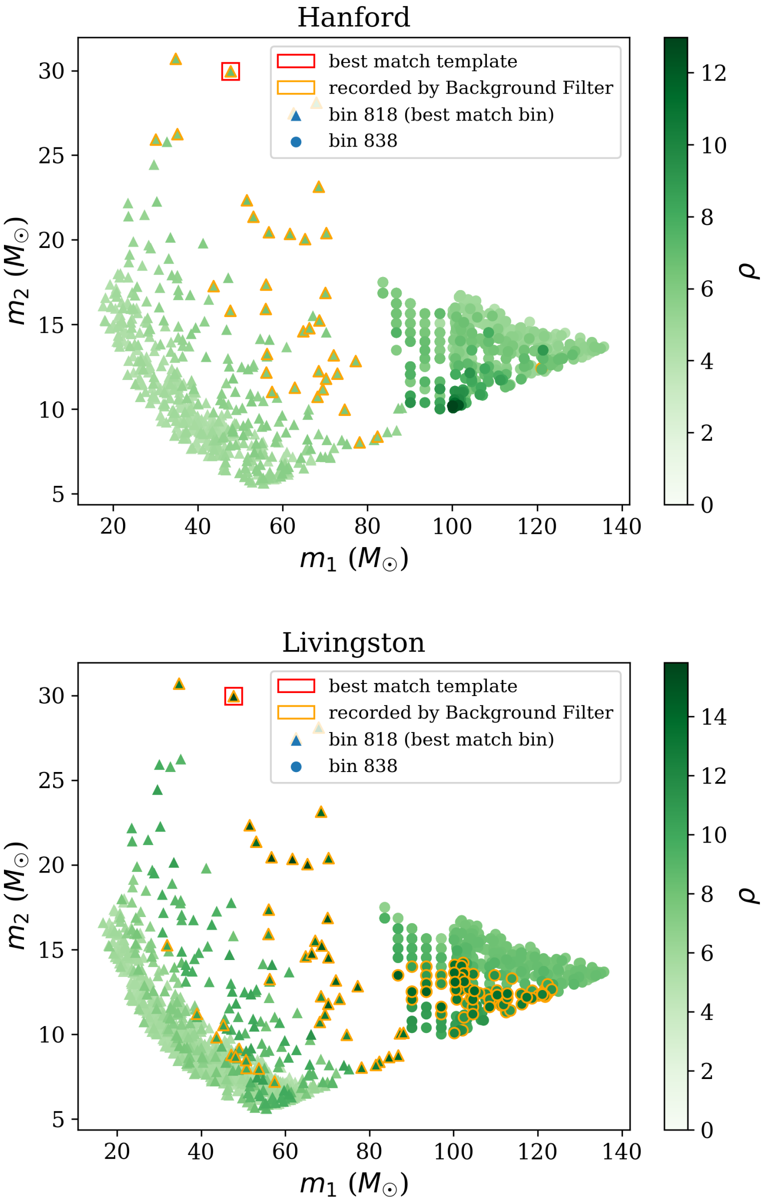


Figure 5.1. An example of an event (GW200129_065458) having templates with high match in multiple template bins. Bin 818 has the best match with the GW candidate, and recovers it in both Hanford and Livingston as a coincidence. Bin 838 has a lower match than bin 818, causing it to recover the candidate as a Livingston single. This will lead to the candidate being added to the $\rho - \xi^2$ background histogram of bin 838, causing signal contamination for bin 838. This is shown in Fig. 5.2. The events passing the ρ and ξ^2 constraints, and hence recorded by the Background Filter are outlined in orange.

5.3.1 Recording events

The strategy of the Background Filter is to record the events that are likely to have originated from GW signals, and then after verification by the user, subtract them from the background histograms. To associate events with a GW candidate, we need to record the time at which they were found in the data, apart from their $\vec{\rho}$ and $\vec{\xi^2}$ values. This increases the dimensionality of the parameters we need to store, and thus could potentially impact the memory and storage used during analysis. To prevent this, we record events only if they pass certain constraints placed on their ρ, ξ^2, and time parameters.

The ρ and ξ^2 constraints take the form of a bounding box in $\rho - \xi^2$ space, defined by $\rho > 6$ and $\xi^2/\rho^2 < 0.04$. Qualitatively, ξ^2 represents how well the data fits the template, with large values of ξ^2 meaning the data is dissimilar to the template. Since in general, noise events will not fit the template well, they generally have ξ^2/ρ^2 values that are greater than those of signals. As a result, we only expect events originating from GW signals to fall inside the bounding box. This is shown in Fig. 5.1, where most of the high ρ events caused by GW200129_065458 pass the ρ and ξ^2 constraints, and are recorded by the Background Filter. The ρ and ξ^2 constraints are shown on top of a background histogram in Fig. 5.3

The time constraint makes use of the GstLAL online analysis' ability to process data, generate events, assign LRs, and upload them to the Gravitational Wave Candidate Event Database (GraceDB) [35] in near real time. A GW signal can create multiple contaminating events across template bins. Only a small subset gets uploaded to GraceDB, since the events are aggregated within some time window across bins before uploading [43], and the remaining comtaminating events lie both before and after the uploaded events in time. With this in mind, and in order to account for processing delays during a GstLAL online analysis, the Background Filter keeps a temporary record of events passing the ρ and ξ^2 constraints, which occurred in the last 5000s. When an event is uploaded to GraceDB, the events in the 10s window around it are found from the temporary record of the last 5000s, and are then recorded by the Bakground Filter permanently.

The threshold for uploading an event to GraceDB differs among different GstLAL analyses, but it is often set to False Alarm Rate (FAR) $<$ 1 per hour. That means all events recovered as a single event during coincident time, with $\rho > 6$, $\xi^2/\rho^2 < 0.04$ and falling in a 10s interval around an event with FAR $<$ 1 per hour are recorded by the Background Filter.

The ρ and ξ^2 constraints, and the time constraint work together to ensure that only

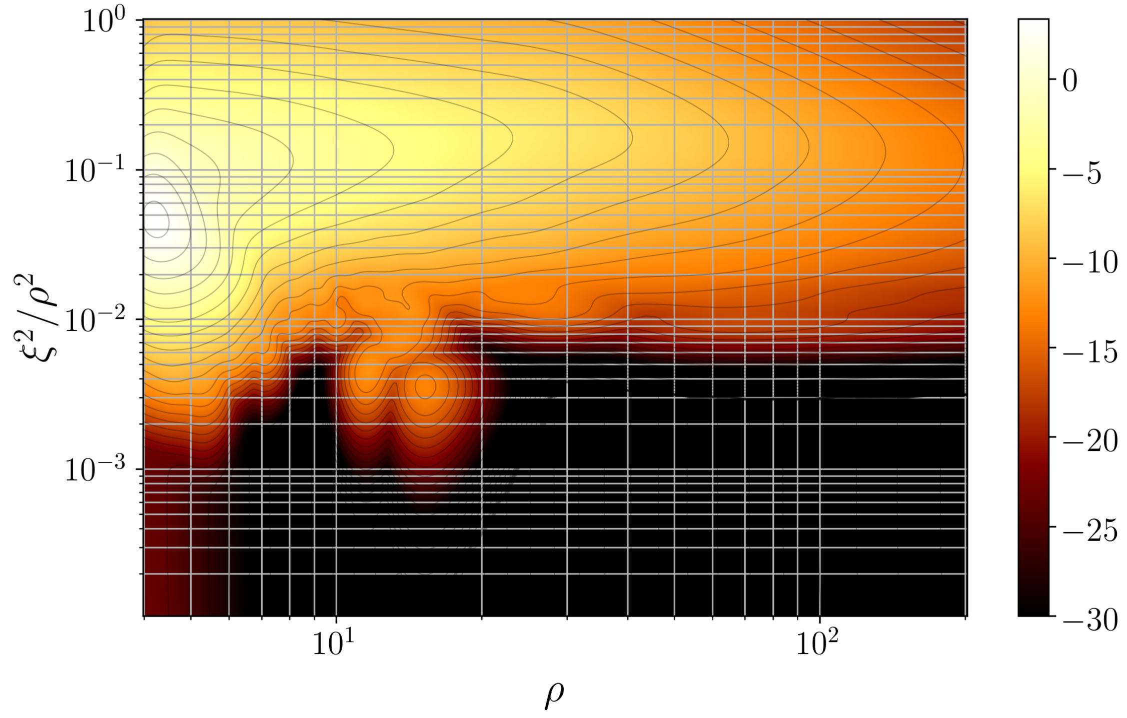


Figure 5.2. An example of signal contamination in a $\rho - \xi^2$ histogram for Livingston. The contamination can be seen as a protrusion to the histogram at (ρ, ξ^2/ρ^2) $\sim$ (15, 0.004), a region ususally occupied exclusively by GW signals. This contamination was caused by GW200129_065458 being recovered as a single event in this template bin, which is not the best match bin for that GW candidate, as demonstrated in Fig. 5.1. Note that kernel smoothing has been applied to this histogram.

events originating from GW signals are recorded by the Background Filter in most cases. As a result, very few events are recorded by the Background Filter, in comparison to the number of events in the background histograms. This ensures that adding the Background Filter to a GstLAL analysis does not affect its memory or disk usage significantly. The choice of these constraints, and their impact on the performance of a GstLAL analysis are discussed in Appendix 5.7.

5.3.2 Removing contamination

As explained in Sec. 5.2, since the $\rho - \xi^2$ noise LR term is calculated by evaluating the probability density function represented by the background histrgrams at the relevant ($\vec{\rho}$, $\vec{\xi^2}$) value, we need the background histograms to accurately reflect the detector noise characteristics for that template bin. As much as possible, we need to take care not to let events originating from signals enter the background histograms. In addition, we must

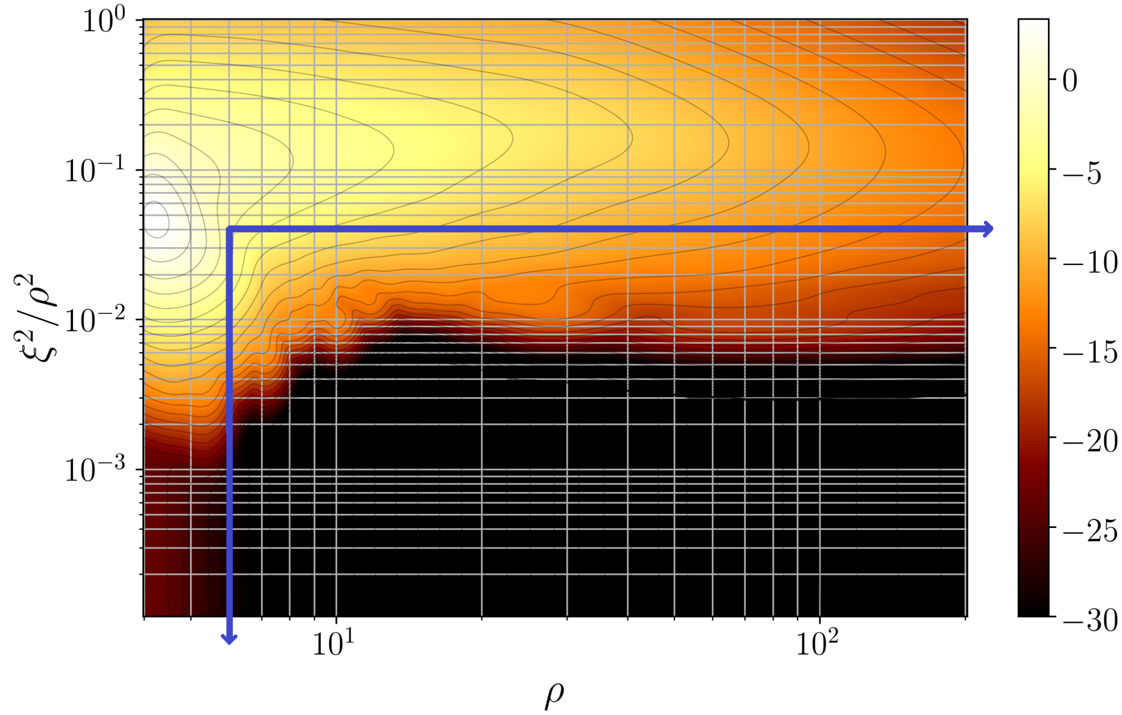


Figure 5.3. The ρ and ξ^2 constraints for recording events. The bottom right area bounded by the blue lines is the area in which the Background Filter records events. If the events also pass the time constraint, the user can choose to remove them from the $\rho - \xi^2$ histogram. The result of doing so, to remove the contamination caused by GW200129_065458 is also shown. The same histogram, without using the Background Filter, and hence with contamination is shown in Fig. 5.2. Note that kernel smoothing has been applied to this histogram.

also make sure that events originating from noise are not removed from the background histograms by the Background Filter. In most cases, the ρ and ξ^2 constraints along with the time constraint are sufficient to ensure only events originating from signals are recorded by the Background Filter.

However, in rare cases, such as when the GstLAL analysis uploads a false positive to GraceDB (also called a “retraction”), these measures might not be enough. Out of an abundance of caution, we leave the decision of which events to remove from the background histograms to the user. At any point during a GstLAL online analysis, the user can choose to inform the analysis which events they are confident are GW candidates. The message is communicated to the analysis in real time using HTTP request methods, with the help of the Python `Bottle` module [134]. Then, out of all the events that had been recorded by the Background Filter previously, it will subtract those which fall within a 10s window of the given candidate, from the background histograms. Thus, any contamination that that candidate could have potentially caused is removed, and the LR of all future events is evaluated using the modified ρ - ξ^2 background histograms.

To summarize, for the Background Filter to subtract an event from the background histograms, three conditions need to be met:

1. The event needs to pass the ρ and ξ^2 constraints
2. The event needs to pass the time constraint
3. The user needs inform the GstLAL analysis that there was a GW candidate nearby the event in time

The first two conditions are sufficient for the Background Filter to record an event and save it to disk, but all three are necessary (and sufficient) for the event to be subtracted from the background histograms. For O4, we have decided that the criteria for informing the analysis of a GW candidate, and hence for removing the background events associated with it are:

1. The event should have a FAR $\leq$ one per 5 months, which is the public alert threshold for significant events [44]
2. The event should not be a retraction, i.e. the GraceDB event should not have the "ADVNO" label applied to it

This criteria was chosen keeping in mind the tradeoff between removing contamination from as many GW candidates as possible, and not removing noise events from the background. This choice and its effects on the sensitivity improvement caused by the Background Filter are discussed in Appendix 5.8.

In Fig. 5.2, signal contamination caused by GW200129_065458 is shown. The same ρ - ξ^2 histogram, but with the Background Filter used to remove that contamination is shown in Fig. 5.3.

5.4 Results

5.4.1 Analysis methods

To test the effect of signal contamination on the sensitivity of a GstLAL analysis, and the ability of the Background Filter to remove the contamination, we analyze a week of O3 data [135], from Apr 18 2019 16:46 UTC to Apr 26 2019 17:14 UTC, in three different ways. First, we perform a control run without any GW signals. Next, to simulate the effect of GW signals, we add “blind injections”. The concept of blind injections, and the set of blind injections that we used are explained in the following subsection. Finally, we perform a “rerank” with the Background Filter enabled, in which LRs are recomputed and significance assignment is done again, but the filtering stage of the GstLAL analysis is taken from the blind injection analysis, since the ρ and ξ^2 values of analyzed events are not affected by the Background Filter, only the LRs and the False Alram Rates (FARs) are. Hence, the rerank corresponds to the case with blind injections present, and the Background Filter being used.

As noted in Sec. 5.3, our policy during O4 for selecting GW candidates to inform the analysis about, is that the candidate should have a FAR $\leq$ one per 5 months. This is also what is done during the rerank, and hence, the Background Filter is only applied to events passing the one per 5 months threshold.

This chunk of data contains two known GW candidates reported in GWTC-2.1 [8] and elsewhere [6, 136], GW190421_213856 and GW190425. However, since we use the Background Filter only on the times of the blind injections, any contamination and subsequent loss in sensitivity caused by either of the two candidates will be present in all three analyses that we perform, and will not affect the evaluation of the performance of the Background Filter.

5.4.2 Simulation Set

Blind injections are simulated GW signals that are added to the data which we analyze and collect background events from (in contrast to regular injections, from which we do not collect background events). We use a set of 868 blind injections distributed across the binary black hole (BBH), binary neutron star (BNS), neutron star-black hole (NSBH), and intermediate-mass black hole (IMBH) parameter spaces. The blind injection set comprises of three subsets, a BNS subset, a BBH subset and a broad subset, with the BNS subset containing half of the total blind injections, and the BBH and broad subsets containing a quarter each. The BNS subset has component masses distributed uniformly from 1 to 3 $M_{\odot}$, and the z-components of dimensionless spin (which are parallel to the orbital angular momentum of the binary) distributed uniformly from -0.05 to 0.05. The BBH subset has component masses distributed uniformly from 5 to 50 $M_{\odot}$ and the z-components of dimensionless spin distributed uniformly from -0.99 to 0.99. The broad subset spans all four parameter spaces mentioned above. It is distributed uniformly in the log of the component masses from 1 to 300 $M_{\odot}$ and has the z-components of dimensionless spin distributed uniformly from -0.99 to 0.99. In addition to the definitions

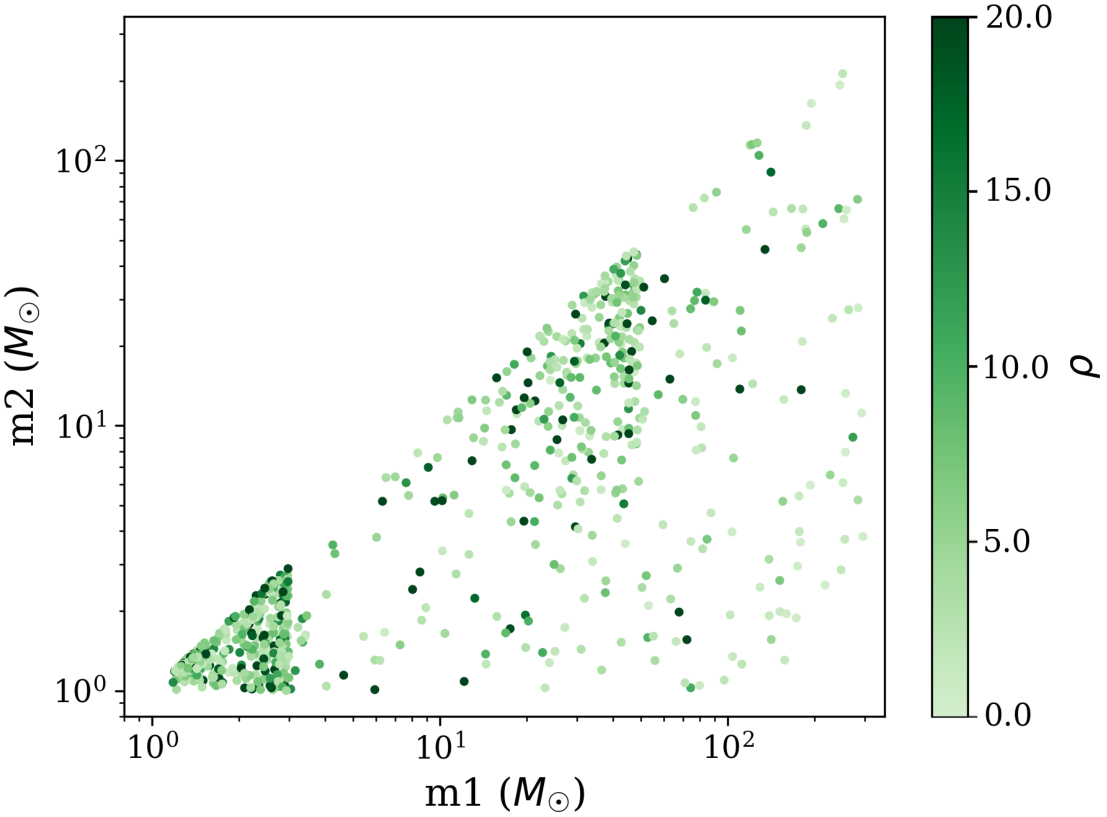


Figure 5.4. The distribution of component masses of the blind injection set, colored by injected ρ. Blind injections are used to replicate the contamination caused by GW signals in the data.

of the BNS and BBH parameter spaces provided above, and the implied NSBH parameter space definitiion, for the purpose of this paper, we shall consider the parameter space with either component mass greater than 50 $M_\odot$ to be the IMBH space. The distribution of the blind injection set in the two component masses can be seen in Fig. 5.4.

A point to note is that even though 868 blind injections may sound high, most of these are too quiet to be recovered, as shown in Fig. 5.4, and hence won't cause any contamination. The result of the analysis shows that only 190 blind injections are recovered with a FAR $\leq$ 1 per 5 months. We will also see later that BNS and NSBH template bins are not affected by signal contamination to a significant degree. As a result, only the BBH and IMBH injections will contribute to contaminating the background. Given the high number of GW candidate events we expect to detect in O4, this is a reasonable representation of the total amount of signal contamination we expect to see.

We also perform an injection campaign to calculate the sensitivity of the analysis, both with and without the application of the Background Filter. The injection set is distributed similarly to the blind injection set, but with a total size of 86,606 injections. It is important to note that the injections and blind injections are analyzed separately, with the blind injections affecting injection recovery only through the background events they add to the ρ - ξ^2 background histograms.

5.4.3 Sensitivity Improvements

In order to estimate the sensitivity of a search, we use the sensitive volume-time (VT) as a measure. The volume that we analyze is determined by the efficiency of recovering injections at a given FAR and redshift, and T is the livetime of the analysis. We calculate VT separately for injections falling in four different chirp mass bins. The first is from 0.5 to 2 $M_{\odot}$, the second from 2 to 4.5 $M_{\odot}$, the third from 4.5 to 45 $M_{\odot}$, and the final one is from 45 to 450 $M_{\odot}$. The reason for calculating VT separately for different mass bins is so that we have an idea about how sensitive the analysis is for different source categories, with the four mass bins roughly corresponding to BNS, NSBH, BBH and IMBH source categories respectively.

Comparing the blind injection analysis with the control run, signal contamination due to the presence of blind injections causes a small ($\sim$ 5%) decrease in VT in the two lowest mass bins, but causes a significant ($\sim$ 20-30%) decrease in VT in the two highest mass bins. This is shown in Fig. 5.5. High mass templates have a greater match with their neighbouring templates, and with themselves across time, as compared to low mass templates. We hypothesize that this causes a single high mass GW signal to be recovered multiple times across template bins and time with suboptimal ρ, leading to more signal contamination in the high mass template bins than in the low mass ones. This is discussed in more detail in Appendix 5.9.

Next, to check the efficacy of the Background Filter in removing contamination, we compare the VT of the rerank to the VT of the control run. Despite the presence of blind injections in the data, The Background Filter mitigates the effect they have on the background histograms, and sensitivities of all four mass bins are close to the same as what they were in the control run. This is shown in Fig. 5.6. This represents a 20-40% increase in the sensitivities of the two high mass bins, in the case of the rerank, as compared to that of the blind injection analysis. We can conclude that the Background Filter is successful in removing close to all of the contamination that the blind injections cause. Since the number of blind injections we used was a high estimate of the number of GW events we expect to see in O4, this means that by using the Background Filter,

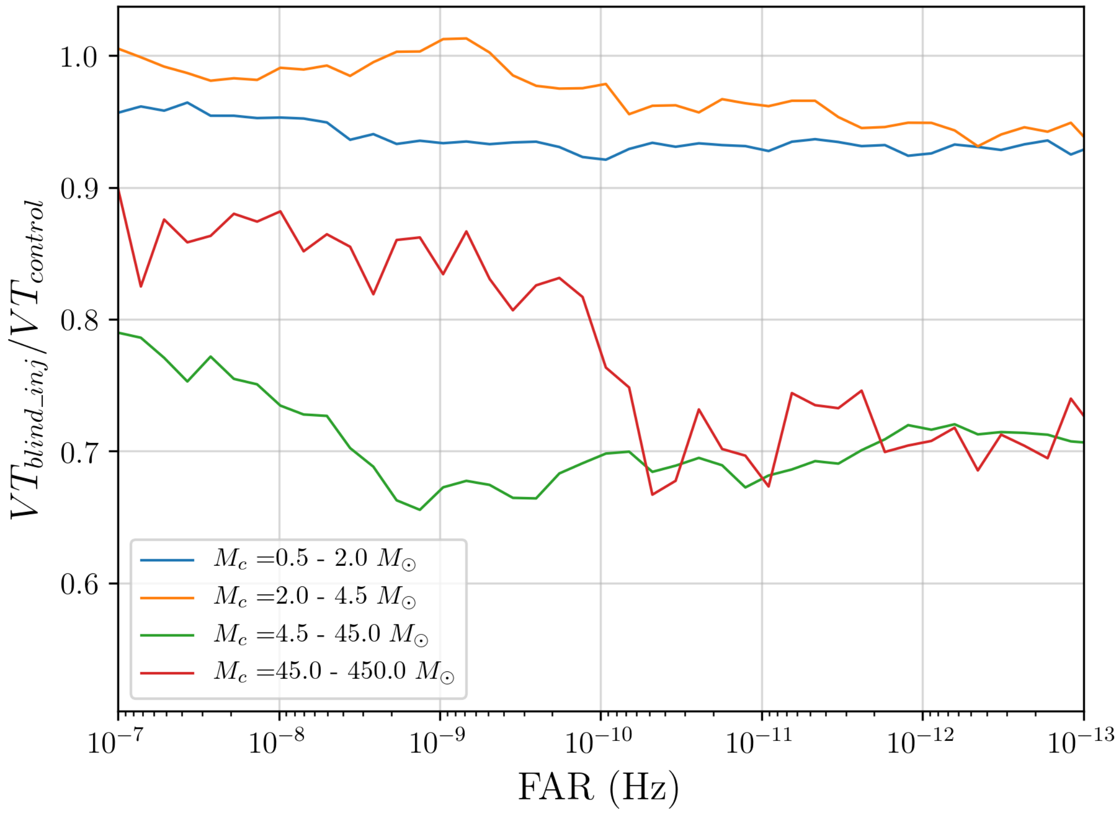


Figure 5.5. The decrease in VT caused by signal contamination due to the presence of blind injections in the data. The two highest mass bins are the most affected. The presence of GW signals will also have a similar effect.

we do not expect signal contamination to be a significant problem during O4. To test our readiness for O4, GstLAL has participated in a mock data challenge, where an online analysis is run over forty days of O3 data [43]. This chunk of data contains 9 GW candidates. The Background Filter was deployed in this analysis, and applied to all 9 candidates. It was able to remove all instances of signal contamination we had previously seen, as verified by visual inspection.

5.5 Conclusion

GstLAL constructs ρ - ξ^2 background histograms to calculate the $P\left(\vec{\rho}, \vec{\xi^2} \mid t_{\mathrm{ref}}, \bar{\theta}, \mathcal{H}_{\mathrm{n}}\right)$ term in the likelihood ratio. However, GW signals in the data can cause the background histograms to be incorrectly constructed. This is called signal contamination, and it

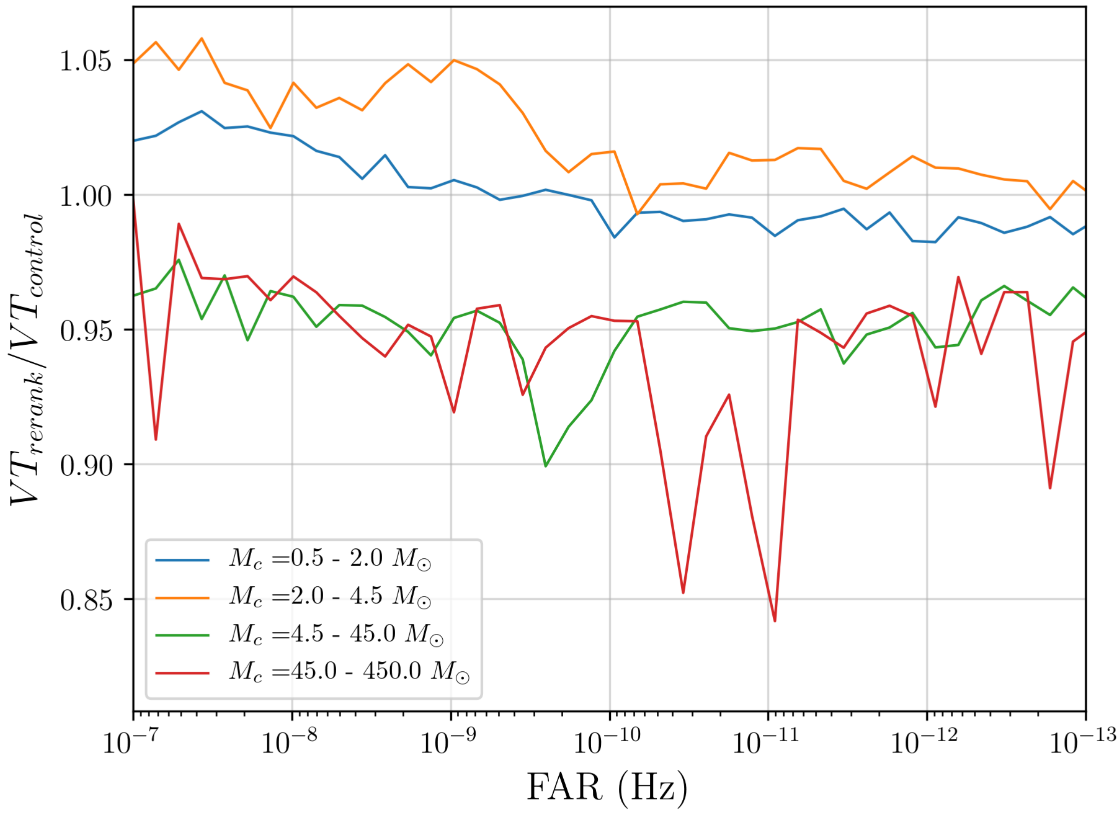


Figure 5.6. The VT of the rerank, which has blind injections with the Background Filter applied, compared to that of the control run, which has neither. The fact that all four lines are close to 1 tells us that the Background Filter is successful in removing nearly all of the contamination caused by the presence of the blind injections in the data. The peaks and dips in the highest mass bin curve are explained by the smaller number of injections in this bin as compared to other bins, leading to greater variance.

leads to the sensitivity of the GstLAL analysis being lowered.

The Background Filter is a novel way to remove the contamination. It records the events that populate the background histograms which satisfy two constraints. The first is that the event must fall in an area in ρ - ξ^2 space consistent with GW signals. The second is that it must fall in a 10s window around a significant event. The user then identifies which of the significant events originate from GW signals. The user communicates this to the GstLAL analysis in real time, and then the events recorded by the Background Filter corresponding to the times identified by the user are subtracted from the background histograms. Thus, signal contamination is removed from the background histograms.

To test the efficacy of the Background Filter, we ran a GstLAL analysis over a week of O3 data, with simulated gravitational-wave signals injected into the data. We found that signal contamination primarily affects the high mass bins. The sensitivity of these bins decreased by 20-30% due to the presence of the gravitational-wave signals. By applying the Background Filter, we were able to increase the sensitivity close to what it was without the injected gravitational-wave signals. This shows that the Background Filter is effective in removing nearly all the signal contamination. With a high rate of gravitational-wave events expected during O4, the Background Filter will be an important tool in improving the sensitivity of the GstLAL analysis.

5.6 Acknowledgements

The authors are grateful for computational resources provided by the LIGO Laboratory and supported by National Science Foundation Grants PHY-0757058 and PHY-0823459. This material is based upon work supported by NSF's LIGO Laboratory which is a major facility fully funded by the National Science Foundation. LIGO was constructed by the California Institute of Technology and Massachusetts Institute of Technology with funding from the National Science Foundation (NSF) and operates under cooperative agreement PHY-1764464. The authors are grateful for computational resources provided by the Pennsylvania State University's Institute for Computational and Data Sciences (ICDS) and by the California Institute of Technology, and support by NSF PHY-2011865, NSF OAC-2103662, NSF PHY-1626190, NSF PHY-1700765, NSF PHY-2207728, and NSF PHY-2207594.

This research has made use of data or software obtained from the Gravitational Wave Open Science Center (gwosc.org), a service of LIGO Laboratory, the LIGO Scientific Collaboration, the Virgo Collaboration, and KAGRA. LIGO Laboratory and Advanced LIGO are funded by the United States National Science Foundation (NSF) as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain. KAGRA is supported by Ministry of Education, Culture, Sports, Science and Technology

(MEXT), Japan Society for the Promotion of Science (JSPS) in Japan; National Research Foundation (NRF) and Ministry of Science and ICT (MSIT) in Korea; Academia Sinica (AS) and National Science and Technology Council (NSTC) in Taiwan.

5.7 Appendix A: Choice of constraints, and their impact on performance

With the constraints described in Sec. 5.3, the Background Filter does not consume too many resources. When looking at a month-long GstLAL analysis, we found that on average, it adds $\sim$ bytes to kilobytes to the data products stored by a GstLAL analysis for every template bin. We didn't see any significant increase to the memory used by the GstLAL analysis either. Fig. 5.6 shows that with these constraints, the Background Filter is effective in removing close to all contamination.

To check if there is any improvement to the sensitivity upon loosening the ρ and ξ^2 constraints, we performed the same analysis as described in Sec. 7.3, but with the ρ and ξ^2 constraints changed to record events with $\rho > 6$ and $\xi^2/\rho^2 < 0.4$. This broader bounding box for recording events did not have any noticeable effect on the sensitivity. This is shown in Fig. 5.7. However, loosening the constraints did increase memory usage of the GstLAL analysis to a noticeable degree.

We do not expect that loosening the $\rho > 6$ constraint or the time constraint would increase sensitivity, since the extra events collected by changing these constraints would be no more significant than noise. This discussion, along with Fig. 5.6 show us that the existing constraints used by the Background Filter satisfy all our requirements.

5.8 Appendix B: Criteria for removing events from the background, and its effect on sensitivity

In Sec. 5.3, we saw that in order for an event to be removed from the background, it needs to pass the ρ and ξ^2 constraints, the time constraint, and the user needs to inform the analysis that there was a GW candidate at the time of the event. This last condition was implemented as an additional check that the event being removed does actually originate from a GW candidate. For the background histograms to accurately model the $\rho - \xi^2$ noise LR term, only events originating from noise must populate the $\rho - \xi^2$ histograms. Events originating from GW candidates entering the histograms, and events

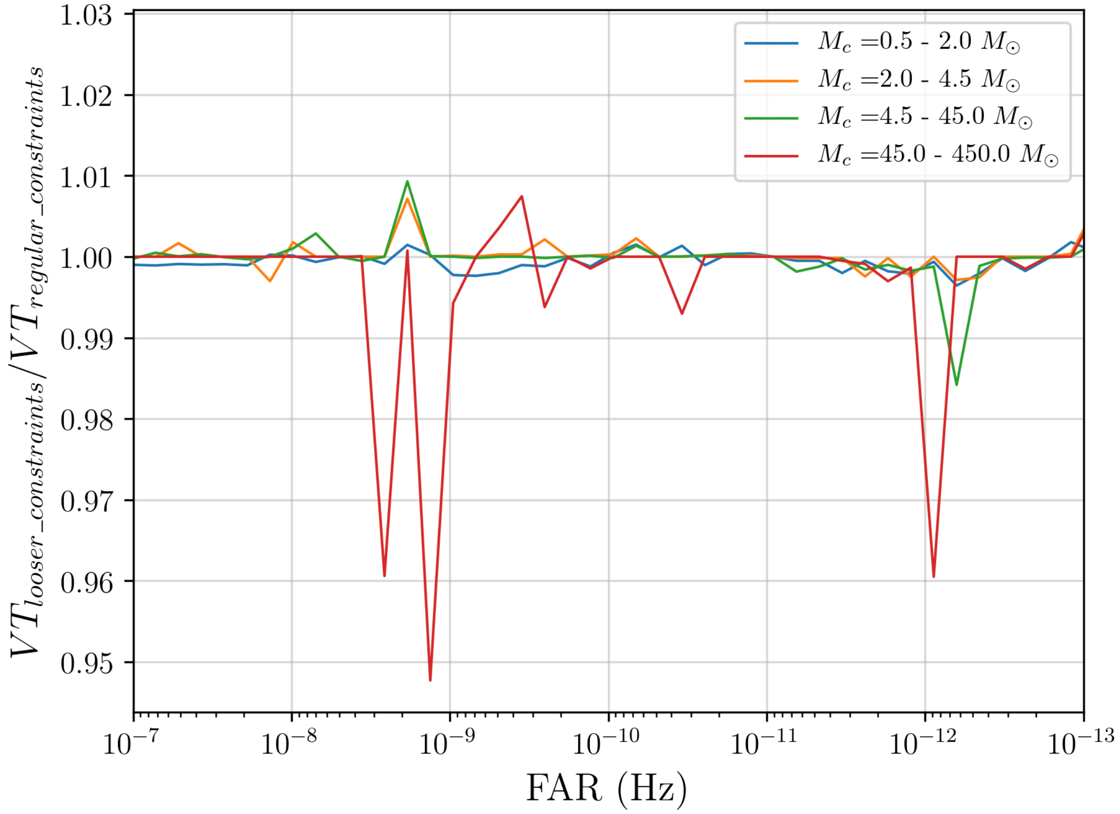


Figure 5.7. The VT with the Background Filter recording events with looser constraints, as compared to the VT with the Background Filter recording events with the regular constraints. The fact that the VT ratios for all four mass bins are close to 1 shows that loosening the constraints does not improve sensitivity. The peaks and dips in the highest mass bin curve are explained by the smaller number of injections in this bin as compared to other bins, leading to greater variance. Both analyses included the 868 blind injections described in Sec. 7.3. The regular constraints are described in Sec. 5.3, whereas the looser constraints are described in Appendix 5.7.

originating from noise being removed from the histograms will both cause the $\rho - \xi^2$ noise LR term to not be evaluated correctly, and will cause a lowering of sensitivity.

As a result, it's important to choose the criteria for informing the analysis of a GW candidate correctly. For O4, we have chosen this criteria to be that the candidate has a FAR $\leq$ 1 per 5 months, and that it is not retracted. The FAR threshold is low enough that it is highly unlikely that the candidate is not astrophysical in origin without being a retraction, while simultaneously being high enough to not exclude too many contaminating (and hence, loud) GW candidates.

To test this, we set up a rerank similar to the one described in Sec. 7.3, but instead

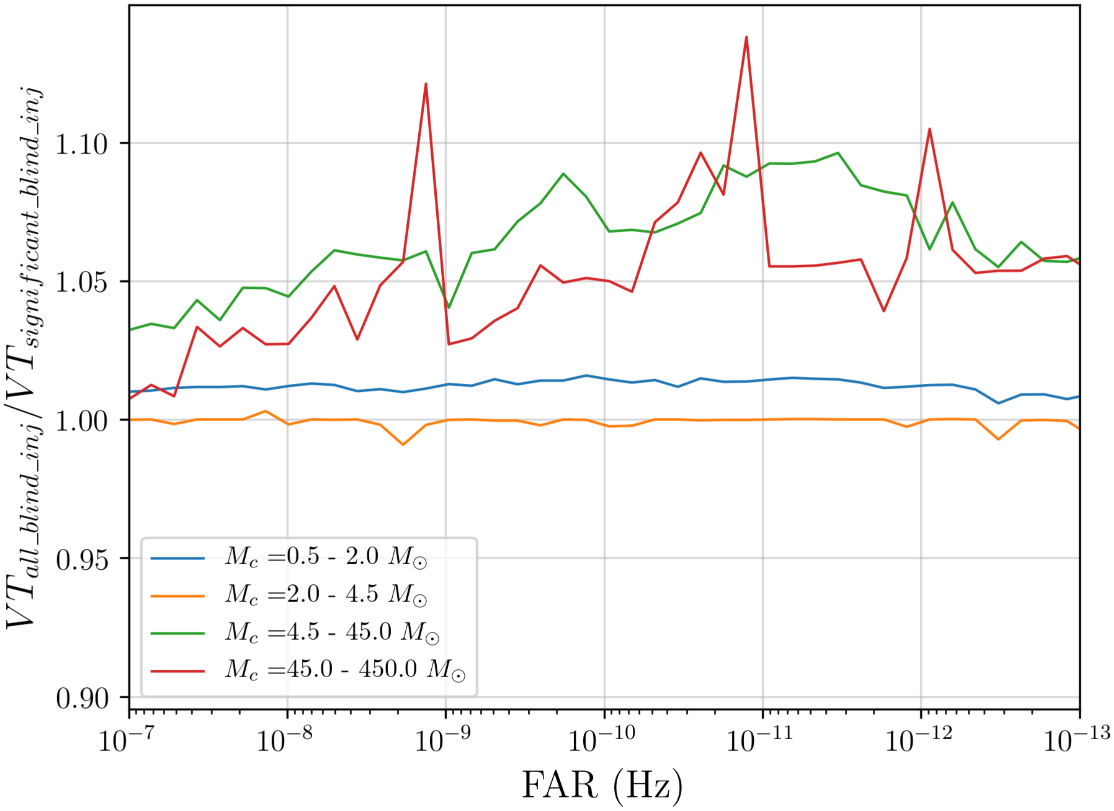


Figure 5.8. The VT with the Background Filter removing contamination from all blind injections, as compared to the VT with the Background Filter removing contamination from blind injections with FAR $\leq$ 1 per 5 months. The first represents the best-case performance of the Background Filter, whereas the second represents the current (O4) performance, designed to prevent noise events from being removed from the background. This graph shows that the current system recovers almost all of the lost sensitivity due to signal contamination.

of applying the Background Filter to blind injections with FAR $\leq$ 1 per 5 months, we applied it to all blind injections. This rerank represents the biggest improvement in sensitivity possible from the Background Filter. The result is shown in Fig. 5.8. We can see that the FAR threshold of 1 per 5 months works well, and out of the 20-30% lost sensitivity due to signal contamination from the blind injections, it manages to recover most of it (see Fig. 5.5, Fig. 5.6), with only around 5% not being recovered in the heavier mass bins.

5.9 Appendix C: Differing impacts of singal contamination of the sensitivities of template bins

As discussed in Sec. 7.3, signal contamination only causes a 5% decrease in the VT of low mass template bins, such as the BNS and NSBH bins, whereas it causes a 20-30% decrease in the VT of high mass template bins, such as the BBH and IMBH bins. This is despite the fact that there are more blind injections in the low mass parameter spaces than in the high mass ones. We conjecture two reasons for this, the first relating to how the correlation among neighbouring templates changes with mass, and the second relating to how the correlation of a templates with itself across time changes with mass. For the remainder of this section, we shall treat BNS template bins as representative of all low mass bins, and IMBH template bins as representative of all high mass bins.

The "bank correlation function" of a template measures how well it matches with other templates in the template bank. This calculation is similar to how ρ is calculated, except that the match is calcualted between two templates with no time shift between them. To see how the bank correlation function of templates changes with mass, we took 5 BNS template bins (corresponding to $\sim$ 5000 templates), calculated the bank correlation of every combination of templates, and plotted the average bank correlation function in descending order of template match. We then did the same for 5 IMBH template bins. The results are shown in Fig. 5.9. The fact that the BNS bank correlation function drops sharply as compared to the IMBH one, means that there are many IMBH templates, across template bins that can recover a given IMBH GW signal with a lower ρ than the best template, but only a few BNS templates that can recover a given BNS GW signal. This means a high mass GW signal will create many events, increasing the probability of signal contamination. This is not a problem for the GW candidates reported by GstLAL, since "event clustering" [19] ensures that only the best candidate in an 8s window survives.

The "autocorrelation function" of a template measures how well it matches with a time-shifted version of itself, similar to how ρ calculates the match between the data and a time-shifted template. The autocorrelation function of a typical BNS template and a typical IMBH template are shown in Fig. 5.10. The IMBH autocorrelation function has multiple secondary peaks $\sim$ 5-10ms away from the primary one. We conjecture that in the case of an IMBH GW singal with low ρ or in high noise, an IMBH template could recover the signal in different detectors at different times, corresponding to the different peaks in the IMBH autocorrelation function. This would cause the signal to be recovered

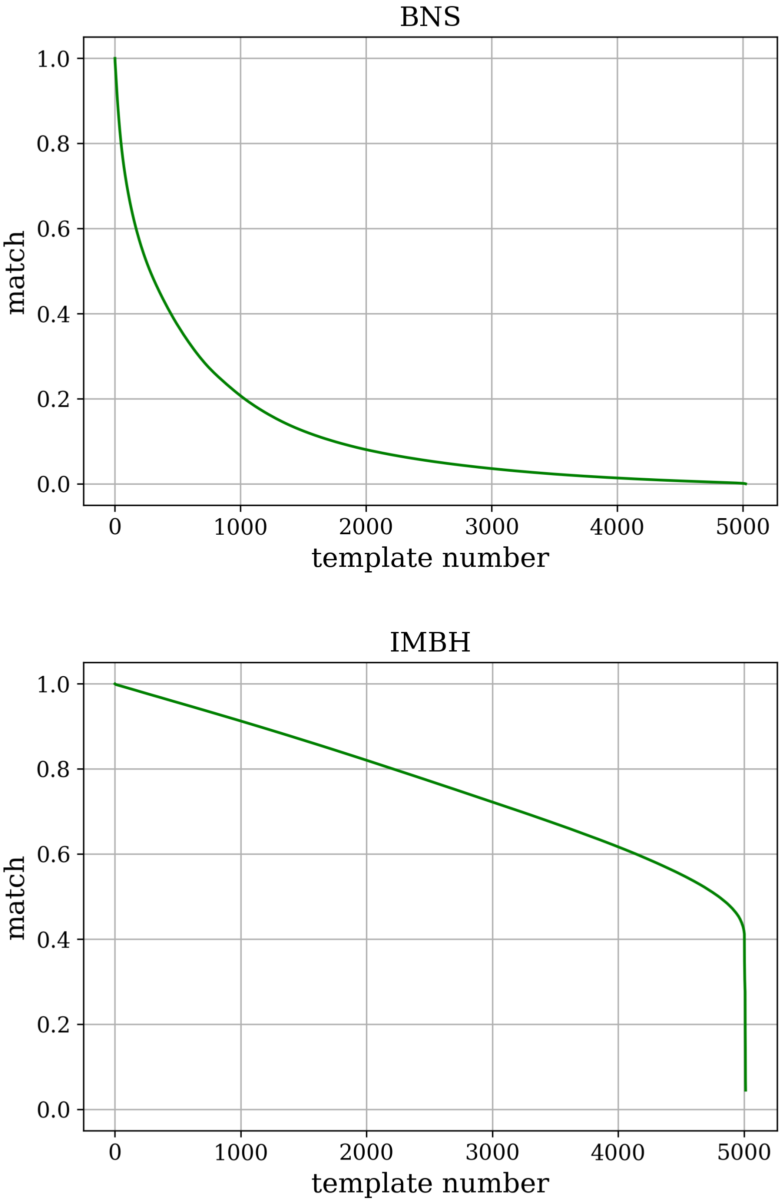


Figure 5.9. The average bank correlation function of a BNS template in descending order of template match, as compared to that of an IMBH template, calculated for the five closest template bins. Since IMBH templates correlate well with other IMBH templates across template bins, an IMBH GW signal will be recovered by multiple template bins, increasing the probability of signal contamination. This is not the case for BNS template bins, and it is more likely a BNS GW signal will be recovered by only one template bin, resulting in fewer cases of signal contamination.

as multiple single detector events, instead of a single coincident one, leading to signal contamination of the high mass template bins. Again, this is not a problem for the GW candidates reported by GstLAL, due to event clustering. Since all the secondary peaks in the autocorrelation function of an IMBH template lie well within an 8s window, multiple single detector events will be clusterd out, and only the best one will survive.

For high mass bins, the bank correlation factor increases the probability of low ρ events getting created by a GW signal, and the autocorrelation factor increases the probability of signal contamination from those events. These two factors compound each other's effect, and as a result, we see a much higher impact of signal contamination in the high mass template bins than in the low mass ones.

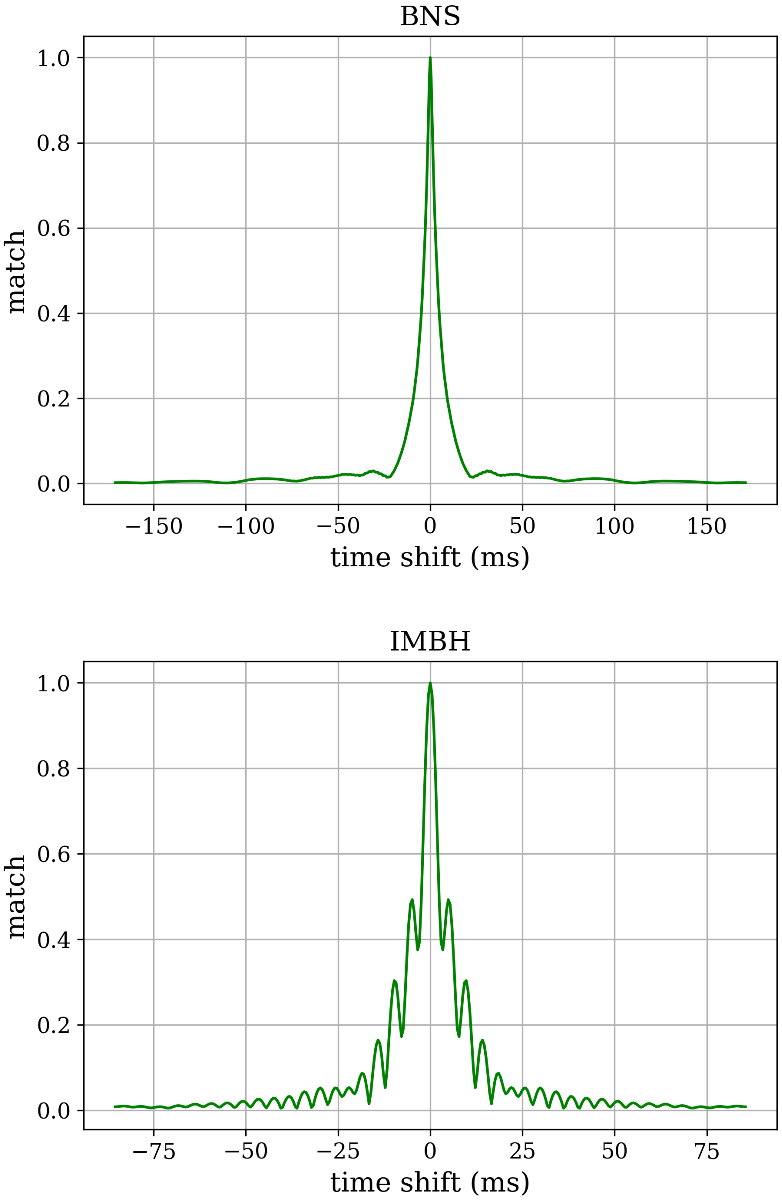


Figure 5.10. The autocorrelation function of a BNS template, as compared to that of an IMBH template. Since there are multiple peaks in autocorrelation function of the IMBH template, a quiet IMBH GW signal could be recovered in different detectors at different times, corresponding to the different peaks in the IMBH autocorrelation function. This will result in the GW singal being recovered as multiple single detector events rather than a single coincident event, which leads to signal contamination. Since the BNS autocorrelation function does not have multiple peaks, signal contamination is less likely for BNS template bins.

Chapter 6
Metric Assisted Stochastic Sampling (MASS) search for gravitational waves from binary black hole mergers

This is a reprint of [137].

We present a novel gravitational wave detection algorithm that conducts a matched filter search stochastically across the compact binary parameter space rather than relying on a fixed bank of template waveforms. This technique is competitive with standard template-bank-driven pipelines in both computational cost and sensitivity. However, the complexity of the analysis is simpler allowing for easy configuration and horizontal scaling across heterogeneous grids of computers. To demonstrate the method we analyze approximately one month of public LIGO data from July 27 00:00 2017 UTC – Aug 25 22:00 2017 UTC and recover eight known confident gravitational wave candidates. We also inject simulated binary black hole (BBH) signals to demonstrate the sensitivity.

6.1 Introduction

Advanced LIGO directly detected gravitational waves (GWs) for the first time in 2015 from the merger of two black holes each about 30 times the mass of our Sun [109]. The second confident binary black hole (BBH) observation came just three months later [138]. Since then, the LIGO and Virgo Collaborations have detected a total of 90 compact binary mergers [6–8, 14], including two neutron star mergers [37, 136] and two neutron star-black hole mergers [139]. LIGO and Virgo have made their data public [140] resulting in several new BBH discoveries by the community [57, 141–145].

Historically, gravitational wave searches for compact binary coalescence have relied on matched filtering [18, 146, 147], with several groups building on matched filtering as the foundation for their algorithms [30, 63, 144, 148–150]. These techniques rely on fixed banks of templates [147, 151, 152] and are known to scale poorly to high dimensional spaces [153]. Stochastic sampling methods were first proposed to address gravitational wave detection in future searches for gravitational waves with LISA [154], but have not been widely used for detection in LIGO and Virgo data. Stochastic sampling techniques are, however, state-of-the art for the estimation of compact binary parameters once detections have been made [47, 155].

In this work we blend aspects of traditional matched filter searches, bank placement techniques, and stochastic sampling to create a new bank-less matched filter search for gravitational waves. While it remains to be seen what the broad applications of these techniques could be, we demonstrate a useful case study here by analyzing LIGO data from the Hanford and Livingston detectors from August 2017 [156] to search for binary black hole mergers. We recover eight known gravitational wave candidates.

6.2 Motivation

Our goal is to develop an *offline* compact binary search pipeline which is designed to detect gravitational waves in archival, LIGO, Virgo, and KAGRA data based on the GstLAL framework [19–21, 157]. We distinguish that an offline analysis has less strict time-to-solution requirements (hours or days) compared to low-latency analysis where the time-to-solution needs to be seconds. We will not strive to reach the time-to-solution needs of low-latency analysis with the algorithm we present here. Our motivation for revisiting offline matched filter detection for gravitational waves is to more easily parallelize and deploy analysis across heterogeneous resources such as multiple concurrent sites on the LIGO and Virgo data grids, the Open Science Grid, campus resources, and commercial clouds. We aim to achieve this by having a simpler workflow than competing pipelines such as GstLAL. We also wish to simplify the setup required to conduct an analysis and to improve usability for new researchers wanting to learn about gravitational wave detection at scale. The intersection of these desires led us to consider new algorithmic approaches to searching the compact binary parameter space.

The Open Science Grid defines criteria for opportunistic computing as an application that "does not require message passing...has a small runtime between 1 and 24 hours...can handle being unexpectedly killed and restarted..." and "...requires running a very large number of small jobs rather than a few large jobs." [158]. Our proposed workflow consists of parallel jobs that each search a small amount of gravitational-wave data from LIGO, Virgo and KAGRA without any interdependency between jobs. To contrast, the current GstLAL analysis workflow consists of a directed acyclic graph with more than ten levels of interdependent jobs. In this new approach, we target a ∼1–12 hour runtime for each job, the use of one CPU core per job, and $\sim$ 2 GB of RAM required per job in order to maximize throughput on opportunistic compute resources. Each job implements a flexible checkpointing procedure allowing work to be periodically saved.

6.3 Methods

In this work, we will conduct a matched-filter search for binary black holes with the goal of identifying the maximum likelihood parameters for candidate events over 4s coalescence-time windows using an analysis that foregoes the use of a pre-computed template bank and instead employs stochastic sampling of the binary parameter space. Our workflow consists of two stages. The first stage executes N parallel jobs that conduct the bulk of the cpu-intensive work - in this study, this first stage consisted of 2974 such jobs. The results of these parallel jobs are returned to a single location at which point a second stage is run to combine results, assess candidate significance, estimate the search sensivity and visualize the results. This second stage requires significantly lower computing power than the first stage, but is I/O intensive and is designed to be run potentially on local resources after grid jobs have completed.

In stage one, we begin by reading in gravitational wave data from each observatory. Next, we measure the data noise power spectrum and whiten the data using the inferred spectrum. We then stochastically sample the data by proposing jumps governed by a parameter space metric as described in Sec. 6.3.4. For each jump, we generate the appropriate template waveform and then compute the matched filter signal-to-noise ratio (SNR) over a 6s stretch of time using 122s of data per calculation.

Within a 4s time window, we identify peaks in the matched filter output, known as triggers, for each detector that is being analyzed. For each collection of triggers, we perform signal consistency checks [19], and calculate a likelihood ratio ranking statistic [24]. If the new sample has a larger SNR than the previous sample, it is stored - otherwise a new jump is proposed.

A local estimate of the noise background is obtained by forming synthetic events from disjoint windows. This causes the time and phase difference between detectors of a single background event to be uniformly distributed, which is what we expect from noise events. This is a somewhat hybrid approach between the time-slide method [149] and sampling methods [25] already employed in GW searches. The second stage gathers the candidate events, results of the simulated GW search, and the background samples to produce a final summary view of the analysis results. In order to estimate the sensitivity of our methods to detecting gravitational waves, we conduct a parallel analysis over the same data with simulated signals added and repeat the same process as described above.

The remainder of this section describes key elements of our methods in more detail.

6.3.1 Data

We assume a linear model for the gravitational wave strain data [146], $\mathbf{s}$, which is a vector of discretely sampled time points for a gravitational wave detector, j,

$$\mathbf{s}_j = \mathbf{h}(\vec{\theta}_j) + \mathbf{n}_j, \tag{6.1}$$

where: $\mathbf{h}(\vec{\theta}_j)$ is an unknown gravitational waveform accurately modeled as a function of $\vec{\theta}_j = \{m_1, m_2, a_{1z}, a_{2z}, t_j, \phi_j, A_j\}$[1] with m_1, m_2 being the component masses, a_{1z}, a_{2z} being the orbital-angular-momentum-aligned component spins and t_j, ϕ_j, A_j being the time of coalescence, phase of coalescence, and amplitude, all of which depend on exactly where the binary is with respect to the jth gravitational wave detector. $\mathbf{n}_j$ is a realization of detector noise. As a concrete example, in this work each job analyzes 800s stretches of data, divided into 4s windows sampled at 2048 Hz. Thus, after including the Fourier transform block length (124s), the dimension of each vector in the work described in this manuscript is 262144 sample points. In addition, each job also contains start padding (128s), and stop padding (32s). The templates have at least 6s of zero padding, which makes their length no more than 122s.

We assume that the noise samples are entirely uncorrelated between the gravitational wave observatories, but that the signals are correlated between observatories. In fact, we make the simplifying assumption that the gravitational waveform is identical between detectors except for an overall amplitude, A_j, time shift, Δt_j, and phase shift, $\Delta \phi_j$,

[1] These parameters are adequate to describe the measurable gravitational wave parameters for a non-precessing, circular binary black hole system with only 2-2 mode emission in a single gravitational wave detector

[148],

$$\mathbf{h}(\vec{\theta}_j) = \Re\left(\mathcal{F}^{-1}\left[A_j e^{2\pi i f \Delta t_j + \Delta\phi_j} \mathcal{F}\left[\mathbf{h}(\vec{\lambda}, t, \phi)\right]\right]\right), \tag{6.2}$$

where $\mathcal{F}[\dots]$ denotes the unitary Fourier transform, and $\vec{\lambda} = \{m_1, m_2, a_{1z}, a_{2z}\}$.

The exact realization of noise, $\mathbf{n}$, is not possible to predict, but we will assume it is well characterized as a multivariate normal distribution with a diagonal covariance matrix in the frequency domain, i.e., that it is stationary. However later on, particularly in Sec. 6.3.4.6, we account for the fact that the data is often not stationary.

6.3.2 Spectrum estimation and whitening

We rely on the same spectrum estimation methods as described in [19]. Namely we use a median-mean, stream-based spectrum estimation technique that adjusts to changes in the noise spectrum on $\mathcal{O}(\text{min})$ time scales. The data are divided into 8s blocks with 6s overlap and the spectrum, $\mathbf{S}_n$ is estimated by windowing the input blocks with 2s of zero-padding on each side of the window. Since we analyze only 800s of data per job, we use a fixed spectrum over the job duration.

From here forward, we will work in a whitened basis for the data, namely that

$$\mathbf{s}_j \rightarrow \mathcal{F}[\mathbf{s}] \circ (\mathbf{S}_n)^{-1/2}, \tag{6.3}$$

which implies that all components of $\mathbf{s}$ are transformed by the inverse noise amplitude spectrum. Therefore, if the amplitude of $\mathbf{h}$ is zero, $\mathbf{s}$ has components that satisfy $p(s_i) = (2\pi)^{-1/2}\, e^{-s_i^2/2}$ with $\langle s_i, s_j \rangle = \delta_{ij}$. In this whitened data basis, an inner product between two vectors is the dot product $\mathbf{u} \cdot \mathbf{v}$, and unit vectors are denoted as $\hat{\mathbf{u}}$. We adopt a normalization such that $\mathbf{h} \cdot \mathbf{h} = 1$ and $\langle \mathbf{n} \cdot \mathbf{n} \rangle = \dim \mathbf{n}$. With these choices the SNR is given by $\rho(\vec{\theta}_j) = \mathbf{h}(\vec{\theta}_j) \cdot \mathbf{s}_j$. We can evaluate the SNR for the unknown phase and time of coalescence by defining a complex SNR

$$\begin{aligned}\rho(\vec{\lambda}, t_j, \phi_j) = {} & \mathcal{F}^{-1}\left[\mathbf{h}(\vec{\lambda}) \cdot \mathbf{s}_j\right] \\ & + i\mathcal{F}^{-1}\left[\mathbf{h}(\vec{\lambda}, \pi/2) \cdot \mathbf{s}_j\right],\end{aligned} \tag{6.4}$$

which is a *valid* matched-filter output for a duration of time equal to the length of the data minus the length of the template. With at least 6s of zero-padding, the template length is 122s, and with each window using 128s of data, the matched-filter output is valid for a duration of 6s.

6.3.3 Simulation capabilities

We use the GstLAL data source module [157], which provides an interface into the LAL Simulation package [159]. By providing a LIGO-LW XML format document containing simulation parameters, we can inject simulated strain into each of the currently operating ground-based gravitational wave detectors, LIGO, Virgo, KAGRA and GEO-600. When operating the pipeline in a simulation mode, gravitational wave events are reconstructed around a ±2s interval around the GPS second of the geocentric arrival time of the gravitational wave peak strain.

6.3.4 Parameter space sampling

The gravitational wave parameter space is explored stochastically, with Gaussian jump proposals and refinement steps that gradually reduce the jump size as the peak in SNR is identified. We will refer to this procedure as "sampling". Our proposal distribution has a covariance matrix that depends on the location in the parameter space and the refinement level. It relies on computing the parameter space metric, $\mathbf{g}$ [147], which is described more in the next section. We define a sequence of two parameters that control how the sampling is done, namely σ_k, which controls the jump size and N_k which controls the number of samples to reject at each level, k, before moving on to the next. How exactly to define these parameters is certainly a topic for future research. Our choices here were determined empirically for the particular search we have done. We define,

$$\sigma_k = 10^{1-k}, \tag{6.5}$$

$$N_k = \begin{cases} 500 & (k = 0) \\ 100 & (k > 0) \end{cases}, \tag{6.6}$$

for $k = 0 \ldots n$ where k is terminated based on the mismatch as in step 7 below. We define the characteristic jump proposal distance as,

$$\delta_k(\vec{\lambda}) := \sigma_k |\mathbf{g}(\vec{\lambda})|^{1/8}, \tag{6.7}$$

where $\vec{\lambda}$ is the set of intrinsic parameters as defined before, and δ_k^2 is the template mismatch.

Gravitational waves are searched over 4 s *windows* of coalescence time using the following procedure.

1. Establish a bounding box in the physical parameter space to search over
2. Pick a starting parameter point somewhere in the middle of the parameter space. We use the approximate expression for template count in [147] to estimate a good central point.
3. Evaluate the SNR at this point and set a counter to zero.
4. Sample from a sampling function $\Theta(\delta_k, \vec{\lambda})$, which is described in detail below in Sec. 6.3.4.4.
5. Check that the new point lies within boundaries established in step 1 and apply any constraint functions. If the point fails to fall within the constraints, go to step 4.
6. Evaluate the SNR at the new point. If the point has a higher SNR than the previous sample, update the sample and reset the counter to zero. If the point has a lower SNR, increment the counter.
7. If the counter is less than N_k, go back to step 4. If the counter is greater than or equal to N_k, check δ^2, where δ^2 is the template mismatch between the current and previous sample point. If $\delta^2 < 0.1$, terminate the sampling. Otherwise, increment k, reset the counter, and proceed to step 4.

6.3.4.1 Computation of the binary parameter space metric

We define the match between adjacent compact binary waveforms in the space of intrinsic parameters as:

$$m(\vec{\lambda}, \vec{\lambda} + \vec{\Delta\lambda}) = \max_{\phi_c, t_c, A} \left[\hat{\mathbf{h}}(\vec{\lambda}) \cdot \hat{\mathbf{h}}(\vec{\lambda} + \vec{\Delta\lambda}) \right]. \tag{6.8}$$

where the maximum is over extrinsic parameters $\{t_c, \phi_c, A\}$. Note that $m(\vec{\lambda}, \vec{\lambda}) = 1$. We also introduce a shorthand for computing the match along a deviation in only one coordinate as:

$$m(\vec{\lambda}, \vec{\lambda} + \vec{\Delta\lambda}_i) = \max_{\phi_c, t_c, A} \left[\hat{\mathbf{h}}(\vec{\lambda}) \cdot \hat{\mathbf{h}}(\vec{\lambda} + \vec{\Delta\lambda}_i) \right], \tag{6.9}$$

where it is assumed that $\Delta\vec{\lambda}_i$ is nonzero only along a given coordinate direction.

It has previously been shown [147] that is possible to derive a metric on the space of intrinsic parameters describing compact binary waveforms by expanding our definition of the match locally e.g. about $\Delta\lambda = 0$ as follows,

$$m(\vec{\lambda}, \vec{\lambda} + \vec{\Delta\lambda}) \approx 1 + \frac{1}{2} \frac{\partial^2}{\partial \Delta\lambda_i \partial \Delta\lambda_j} m(\vec{\lambda}, \vec{\lambda} + \vec{\Delta\lambda}) \Big|_{\vec{\Delta\lambda}=0} \Delta\lambda_i \Delta\lambda_j \tag{6.10}$$

which suggests the metric

$$g_{ij}(\vec{\lambda}) = -\frac{1}{2} \frac{\partial^2}{\partial \Delta\lambda_i \partial \Delta\lambda_j} m(\vec{\lambda}, \vec{\lambda} + \vec{\Delta\lambda}) \Big|_{\vec{\Delta\lambda}=0}, \tag{6.11}$$

The mismatch between templates, $\delta^2 = 1 - m$ becomes

$$\delta(\vec{\lambda}, \vec{\Delta\lambda})^2 \approx \vec{\Delta\lambda}^T \, \mathbf{g}(\vec{\lambda}) \, \vec{\Delta\lambda}, \tag{6.12}$$

In this work, the components of the metric are evaluated with second-order finite differencing,

$$g_{ii}(\vec{\lambda}) = -\frac{1}{2} \left[\frac{m(\vec{\lambda}, \vec{\lambda} + \Delta\vec{\lambda}_i) + m(\vec{\lambda}, \vec{\lambda} - \Delta\vec{\lambda}_i) - 2}{|\Delta\vec{\lambda}_i|^2} \right], \tag{6.13}$$

and

$$\begin{aligned} g_{ij}(\vec{\lambda}) = -\frac{1}{2} \frac{1}{4|\Delta\vec{\lambda}_i||\Delta\vec{\lambda}_j|} \times \Bigg[& \; m(\vec{\lambda}, \vec{\lambda} + \Delta\vec{\lambda}_i + \Delta\vec{\lambda}_j) \\ & - m(\vec{\lambda}, \vec{\lambda} + \Delta\vec{\lambda}_i - \Delta\vec{\lambda}_j) \\ & - m(\vec{\lambda}, \vec{\lambda} - \Delta\vec{\lambda}_i + \Delta\vec{\lambda}_j) \\ & + m(\vec{\lambda}, \vec{\lambda} - \Delta\vec{\lambda}_i - \Delta\vec{\lambda}_j) \Bigg], \end{aligned} \tag{6.14}$$

for the off diagonal terms. However, we use a more efficient formula for the off diagonal terms, in which the number of template evaluations is the same, but the number of match calculations is reduced:

$$\begin{aligned} g_{ij}(\vec{\lambda}) = &-\frac{1}{2}\frac{1}{2|\Delta\vec{\lambda}_i||\Delta\vec{\lambda}_j|}\times \\ &\Bigg[\ m(\vec{\lambda},\vec{\lambda}+\Delta\vec{\lambda}_i+\Delta\vec{\lambda}_j) - m(\vec{\lambda},\vec{\lambda}+\Delta\vec{\lambda}_i) \\ &\ - m(\vec{\lambda},\vec{\lambda}+\Delta\vec{\lambda}_j) + 2 - m(\vec{\lambda},\vec{\lambda}-\Delta\vec{\lambda}_i) \\ &\ - m(\vec{\lambda},\vec{\lambda}-\Delta\vec{\lambda}_j) + m(\vec{\lambda},\vec{\lambda}-\Delta\vec{\lambda}_i-\Delta\vec{\lambda}_j)\ \Bigg] \end{aligned} \tag{6.15}$$

The sampling method described in section 4 below will not make jumps in coalescence time, therefore the time component is projected out [147],

$$g_{ij}(\vec{\lambda}) \to g_{ij}(\vec{\lambda}) - \frac{g_{ti}(\vec{\lambda})\, g_{tj}(\vec{\lambda})}{g_{tt}(\vec{\lambda})}. \tag{6.16}$$

6.3.4.2 Choice of coordinates

We sought out a coordinate system that maps the masses and spins to be in the interval $[-\infty, \infty]$. We also want to choose well measured physical parameters for mass and spin in at least one dimension each. Therefore, we use the following coordinates to evaluate the metric

$$\lambda_1 = \log_{10}\left[\frac{(m_1 m_2)^{3/5}}{(m_1+m_2)^{1/5}}\right] \tag{6.17}$$

$$\lambda_2 = \log_{10}(m_2) \tag{6.18}$$

$$\lambda_3 = \tan\left[\left(\frac{\pi}{2}\right)\left(\frac{a_{1z}m_1 + a_{2z}m_2}{m_1+m_2}\right)\right] \tag{6.19}$$

$$\lambda_4 = \tan\left[\left(\frac{\pi}{2}\right)\left(\frac{a_{1z}m_1 - a_{2z}m_2}{m_1+m_2}\right)\right] \tag{6.20}$$

6.3.4.3 Pathologies of the numerical metric

For certain regions of the parameter space the metric is nearly singular which leads to numerical errors causing a non positive definite matrix. To fix this, we conduct an eigenvalue decomposition of g_{ij}

$$g_{ij} = q_{ik}\, \beta^k\, q^{-1}_{kj} \tag{6.21}$$

We then define a new set of eigenvalues

$$\beta_{\rm min} \equiv \max_k \left[\beta^k\right] \times \epsilon \tag{6.22}$$

$$(\beta^k)' = \begin{cases} \beta_{\rm min}, & \beta^k < \beta_{min} \\ \beta^k, & \text{otherwise,} \end{cases} \tag{6.23}$$

where ϵ is a parameter which we will call the aspect ratio. We define the new metric as

$$g_{ij} \to q_{ik}\,(\beta^k)'\,q^{-1}_{kj} \tag{6.24}$$

In practice we find that sampling is better when we artificially distort the metric by setting $\epsilon = .1$ for the broadest refinement level, and $\epsilon = .0001$ for all other levels, and we have done so in this work, though this should be a direction of future work.

6.3.4.4 Drawing random samples from $\Theta(\delta_k, \vec{\lambda})$

When sampling, we desire to have a jump proposal distribution that effectively probes the space by not making jumps that are either too near or too far. The calculation of the parameter space metric $\mathbf{g}$ enables that. We wish to propose a jump from $\vec{\theta} \to \vec{\theta} + \Delta\vec{\theta}$ such that the expected mismatch is δ^2. The metric described in previous sections only applies to the intrinsic parameters. For the extrinsic parameters, our jump proposal will always choose those values of t and ϕ which maximize the SNR. At every accepted jump point, the metric is calculated locally, which requires 21 template evaluations, including the diagonal and off diagonal terms, as specified in Sec. 6.3.4.1. However, we can afford to calculate coarse versions of the template waveform, since the match we need to calculate is between two adjacent templates. This means the waveform calculation cost is not high. The distance between adjacent templates to calculate the match at, $\Delta\vec{\lambda}_i$ as defined in Sec. 6.3.4.1 is hardcoded, and is the same for all iterations of the sampling procedure.

To facilitate jumping in the intrinsic parameters, we make a coordinate basis transformation in which the new basis has a Euclidian metric. The transformation matrix $\mathbf{M}$ will then be used to transform the coordinates

$$\vec{\lambda}' = \mathbf{M}\vec{\lambda}. \tag{6.25}$$

To solve for $\mathbf{M}$ we rely on the fact that distance is invariant giving

$$\begin{aligned}
\delta^2 &= \vec{\Delta\lambda}^T \, \mathbf{g} \, \vec{\Delta\lambda} && (6.26) \\
&= (\vec{\Delta\lambda}')^T \, \mathbf{g}' \, \vec{\Delta\lambda}' && (6.27) \\
&= (\vec{\Delta\lambda}')^T \, \mathbf{M}^T \mathbf{g} \, \mathbf{M} \, \vec{\Delta\lambda}' && (6.28)
\end{aligned}$$

Setting $\mathbf{g}' = \mathbf{I}$ gives

$$\begin{aligned}
\mathbf{I} &= \mathbf{M}^T \mathbf{g} \, \mathbf{M} && (6.29) \\
\mathbf{M}^{-1} (\mathbf{M}^T)^{-1} &= \mathbf{g} && (6.30) \\
\mathbf{g}^{-1} &= \mathbf{M}^T \mathbf{M} && (6.31)
\end{aligned}$$

The last line implies that we can solve for $\mathbf{M}$ by taking the Cholesky decomposition of the inverse metric tensor. Once obtaining $\mathbf{M}$ we can produce random samples with an expected mismatch by defining,

$$\Theta(\delta_k, \vec{\lambda}) := \delta_k, \vec{\lambda} \to \vec{\lambda} + \delta_k \mathbf{M}^T \vec{\mathcal{N}}, \qquad (6.32)$$

where $\vec{\mathcal{N}}$ is a 4-dimensional vector with random components satisfying $p(\mathcal{N}_i) = \sqrt{1/2\pi} \exp\left[-\mathcal{N}_i^2/2\right]$

6.3.4.5 Parameter space constraints

The previously defined sampling function can produce samples that, while physical, may be outside of the desired search range. We implement a series of user-defined constraints that will reject samples drawn from $\Theta(\delta_k, \vec{\lambda})$. These are:

$\mathbf{m_1}, \mathbf{m_2}, \mathbf{a_1}, \mathbf{a_2}$ The user can specify a bounding-box in component masses and z-component spins. Samples outside this bounding box are rejected.

$\boldsymbol{\eta}$ The user can specify a minimum symmetric mass ratio, $\eta \equiv (m_1 m_2)/(m_1 + m_2)^2$, below which samples will be rejected.

$\mathcal{M}$ The user can specify a chirp mass range outside of which samples will be rejected.

6.3.4.6 Glitch Rejection

Glitches [160, 161] are non-stationarity and non-Gaussian transient noise artefacts of instrumental or environmental origin found in the data. We employ a novel data-driven technique to reject short-duration glitches, using two parameters, the bandwidth, and the effective spin parameter χ. The bandwidth is the standard deviation of frequency weighted by template amplitude. It is defined as [162]:

$$\text{bandwidth}^2 = \frac{\int |\hat{\mathbf{h}}(\vec{\lambda})|^2 f^2 df/\mathbf{S}_n}{\int |\hat{\mathbf{h}}(\vec{\lambda})|^2 df/\mathbf{S}_n} - \left(\frac{\int |\hat{\mathbf{h}}(\vec{\lambda})|^2 f df/\mathbf{S}_n}{\int |\hat{\mathbf{h}}(\vec{\lambda})|^2 df/\mathbf{S}_n}\right)^2 \tag{6.33}$$

whereas χ is defined as:

$$\chi = \frac{m_1 a_{1z} + m_2 a_{2z}}{m_1 + m_2} \tag{6.34}$$

It has been found that short-duration glitches ring up templates which exclusively occupy the low bandwidth-low χ region in bandwidth-χ space, and that this region is not occupied by gravitational wave signals. This is illustrated in Fig. 6.1. As part of the simulation campaign we performed (Refer to Sec. 6.4.3 for details), we found that only 28 injections out of 112526 fell into the glitch region. Minimizing this number by fine-tuning the boundary of the glitch region would be a direction for future work. We define the glitch region as:

$$\text{bandwidth} \times (1 + \chi) \leq 20 \tag{6.35}$$

Any trigger which falls in this region is not considered as a gravitational wave candidate. Similarly, any time-slid background samples falling in the glitch region are eliminated, and not used for background estimation. Triggers are explained in more detail in the next subsection, whereas background estimation is explained in Sec. 6.3.5.

6.3.4.7 Computing the log-likelihood ratio, $\mathcal{L}$

We generally follow the same procedure for ranking candidates as described in [22, 24, 25] with a couple of notable exceptions. First, we only implement a subset of the terms used in the GstLAL-inspiral pipeline – it will be the subject of future work to include more. Second, we approximate some of the data driven noise terms with analytic functions. Third, we adopt a normalization so that for signals, the log likelihood ratio, $\mathcal{L}$ is approximately $\rho^2/2$, where ρ is the network matched filter SNR defined as the squareroot of the sum of the squares of the SNRs found in each observatory. We use the following terms in the log likelihood ratio:

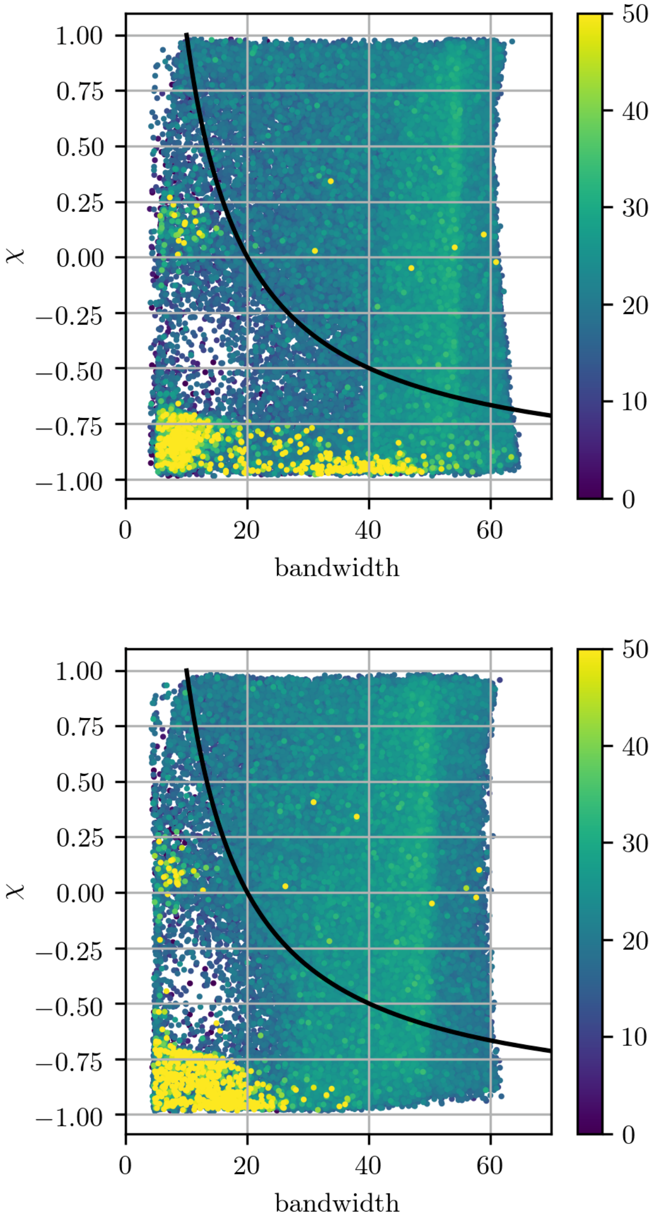


Figure 6.1. Triggers found in one month's data for Hanford (top) and Livingston (bottom), colored by their log-likelihood ratio. All the bright points to the right of the boundary are known gravitational wave candidates, and all those to the left of the boundary are glitches, and so not considered gravitational wave candidates, and not used for background estimation.

$\mathcal{L}(\vec{\rho}, \vec{\xi^2})$ We approximate this term of the log likelihood ratio as

$$\mathcal{L}(\vec{\rho}, \vec{\xi^2}) = \sum_i \mathcal{L}_i(\rho_i, \xi_i^2) \tag{6.36}$$

$$\text{with } \mathcal{L}_i(\rho_i, \xi_i^2) = \rho_i^2 e^{-4x_i^2}/2 - 4x_i^2 \tag{6.37}$$

where $x_i \equiv \max\{0, \xi_i^2 - 1 - 0.0005\rho_i^2\}$ for each of the i detectors which are assumed to be independent. The $-4x_i^2$ term acts as a penalty for high ξ_i^2 values, and helps eliminate glitches.

$\mathcal{L}(\Delta\vec{t}, \Delta\vec{\phi}, \Delta\vec{\rho})$ for this term we follow the procedure in [22] with two changes. We do not include the ρ^{-4} term. We do this because we are not constructing a data driven noise term like GstLAL-inspiral, so it's not necessary to have the corresponding signal term. We also normalize the result to be 0 when only one detector is operating. This is useful for achieving the normalization discussed above. These changes have the effect of making this term ~ 0 for things that are consistent with signals.

$\mathcal{L}(\vec{T})$ this term quantifies the probability of having "triggered" the combination of the gravitational wave detectors in which the event was found and is a function of the detectors' sensitivity. We will describe triggering in more detail below. For example, it is unlikely that only the least sensitive detector would be triggered for a real gravitational wave event, so this term would be negative in that case. This term is complementary to the previous term but accounts for events lacking triggers.

$\mathcal{L}(\vec{D}_H)$ this term quantifies the relative likelihood of detecting an event based on the detector horizon BNS distances, $(D_H)_i$. We normalize to the horizon distance of LIGO Livingston during O3$\sim$ 315 Mpc.

$$\mathcal{L}(\vec{D}_H) = \ln\left(\frac{\max_i\left[\{(D_H)_i\}\right]}{315Mpc}\right)^3 \tag{6.38}$$

The log-likelihood ratio, $\mathcal{L}$ is then given by

$$\begin{aligned}\mathcal{L} = {} & \mathcal{L}(\vec{\rho}, \vec{\xi^2}) + \mathcal{L}(\Delta\vec{t}, \Delta\vec{\phi}, \Delta\vec{\rho}) \\ & + \mathcal{L}(\vec{T}) + \mathcal{L}(\vec{D}_H)\end{aligned} \tag{6.39}$$

For each sample drawn in step 4, we construct a template waveform $\mathbf{h}(\vec{\lambda})$, and filter that waveform against the data in each detector stream producing an SNR time series over a 6s period, including 1s padding on either side. We then find the peak SNR in the middle 4s window in each detector and record the time, phase, SNR, and ξ^2 of each peak, which we call a "trigger". For the collection of triggers, we cycle through every detector combination - for example, if analyzing {H,L,V}, we cycle through {HLV, HL, HV, LV, H, L, V} and evalute the likelihood ratio for each combination. We then keep the maximum $\mathcal{L}$ found over these detector combinations. This is done to mitigate the effect of bad data (noisy data and possibly also glitchy data) in one detector. Hence, triggers are obtained by maximizing SNR over 4s windows, whereas the detectors to be considered for the trigger are obtained by maximizing the likelihood ratio over all possible detector combinations. Note that the SNR maximization for updating the sample discussed in step 6 is a seperate procedure from either of these.

6.3.5 Background estimation

We treat windows recovered as single triggers and windows recovered as coincidences differently while estimating the background. For single trigger windows, the foreground sample itself is used as the background sample representing that window. To estimate the coincident background, we form false coincidences from a given job which analyzes 800s of data in 200 coalescence time windows. To form false coincidences, we shift the windows in time with respect to each other. We then draw samples randomly from all single detector triggers. For each recovered false coincidence, we compute a $\mathcal{L}$ and histogram the result. This process is then repeated 100 times with different time offsets to increase the amount of background we have. This background is given an appropriate weight so that the ratio of singles to coincidences in the background and foreground is the same, as well as to ensure that the background is normalized. Using the $\mathcal{L}$ histogram for the background, false alarm rates (FARs) are assigned to all the triggers. One point to note is that the windows in which we detect events are not used to form combinations so as to not contaminate the background with signals.

6.4 Results

6.4.1 Data set

We analyze public gravitational wave data from LIGO taken from July 27 00:00 2017 UTC – Aug 25 22:00 2017 UTC during advanced LIGO's second observing run. We choose segments of data with a minimum length of 1200s for each of the LIGO detectors. From those segments we form coincident segments. Jobs require 128s of start padding, 32s of end padding and 124s for the Fourier transform block to produce triggers. Thus, each job can analyze a minimum of 288s (which produces triggers for for a single 4s window) and we choose a maximum duration of 1084s to produce 800s of triggers over 200 windows. Jobs are overlapped so that triggers are produced contiguously.

After accounting for the segment selection effects, we analyzed approximately 20.17 days of coincident data.

6.4.2 Search parameter space

We search for gravitational wave candidates with component masses between 0.9–400 $\mathrm{M}_{\odot}$ with z-component spins between -1 and 1. We conduct the matched filter integration between 10–1024 Hz.

6.4.3 Simulation set

In order to ascertain our sensitivity to gravitational wave signals of the type discovered in this data, we conducted a simulation campaign with 112526 simulated signals having a 32 $\mathrm{M}_{\odot}$ mean component mass and standard deviation of 4.0 $\mathrm{M}_{\odot}$ with aligned dimensionless spins up to 0.25 and a maximum redshift of 1 isotropically distributed in location. The injections were distributed uniformly in comoving volume. The red-shifted component mass distribution is visualized in Fig. 6.2. The injection set was specifically created for the BBH parameter space. We do not make any claims about the sensitivity of our pipeline in other lower mass parameter spaces.

6.4.4 Candidate list

Our search results are summarized in Fig. 6.3 and Table 6.1. Results from the entire search are shown in Fig. 6.3. In this plot, we show the observed distribution of all events as a function of $\sqrt{2\mathcal{L}}$, an expression proportional to the SNR, as well as the background

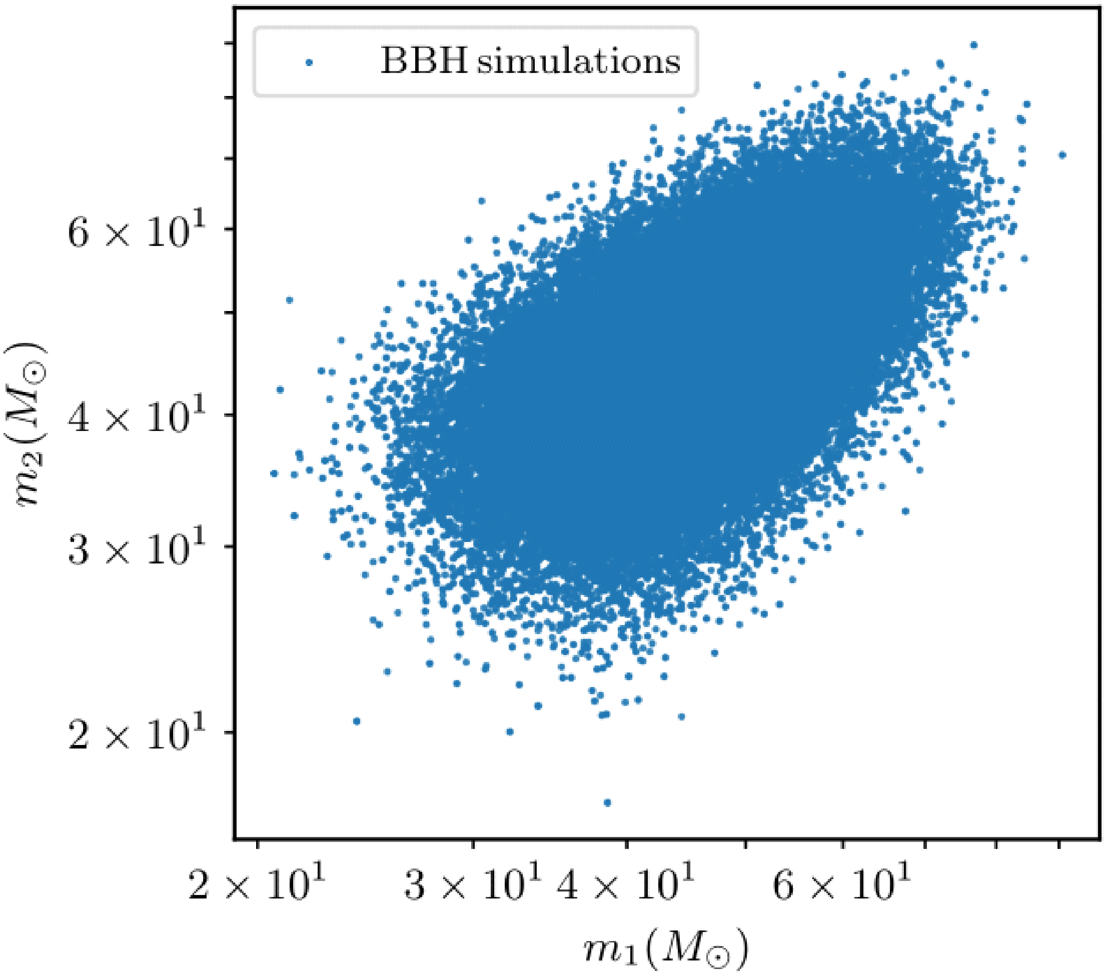


Figure 6.2. Distribution of component masses as measured at Earth for the BBH simulation set.

distribution expected from noise during the same time. The detected events clearly stand out from the expected noise curve at $\sqrt{2\mathcal{L}}$ around 8 which suggests that the extra events at high $\mathcal{L}$ must be signal-like.

In Table 6.1, we report the ten triggers with the smallest FARs. The first five of these events as well as the seventh were previously reported by the LIGO Collaboration and others [14, 142, 163] and labelled GW170817, GW170814, GW170809, GW170823, GW170729, and GW170818. These events are detected confidently with FARs of $5 \times 10^{-3} yr^{-1}$ for the first five, and $4 \times 10^{-2} yr^{-1}$ for the seventh. GW170817 is recovered as a single detector candidate in Hanford, since there's a simultaneous glitch in Livingston, and the resulting high ξ^2 in Livingston causes its log-likelihood ratio to be strongly

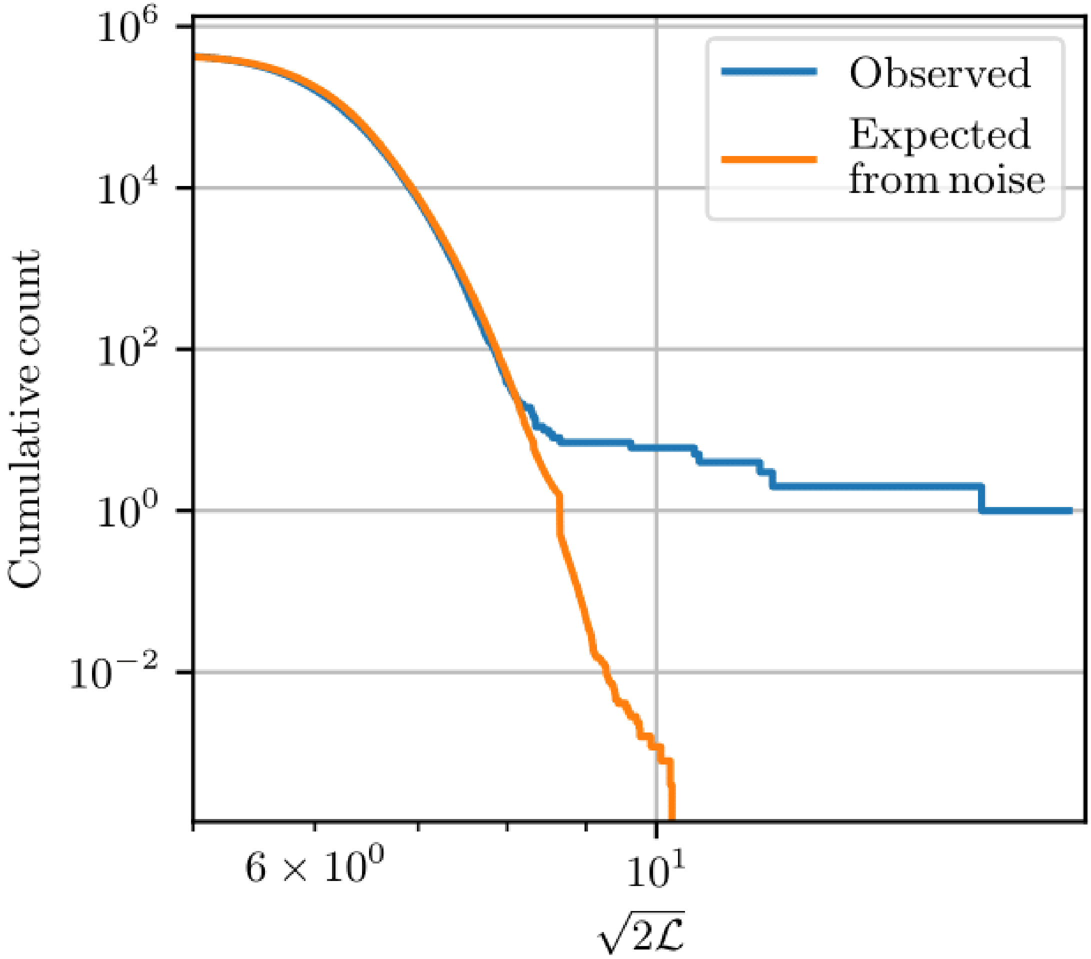


Figure 6.3. Cumulative histograms of our search results as a function of likelihood ratio. The orange line represents the corresponding histogram expected from noise during the same time frame.

penalized. We report many of the components masses of these events outside of confidence ranges reported by the LIGO Collaboration [14]. It is important to note that this is not a contradiction: we are not optimizing the posterior probability distribution, as is done during parameter estimation for the results reported by the LIGO Collaboration. Despite the differences in masses, we are able to recover each trigger to within tens of milliseconds of the reported values by the LIGO Collaboration and are confident they correspond to the respective gravitational wave candidates.

We also recover one binary black hole event, GW170727 previously reported by other groups [142, 163] as well as one, GW170817a reported by Zackay et al [141] which do

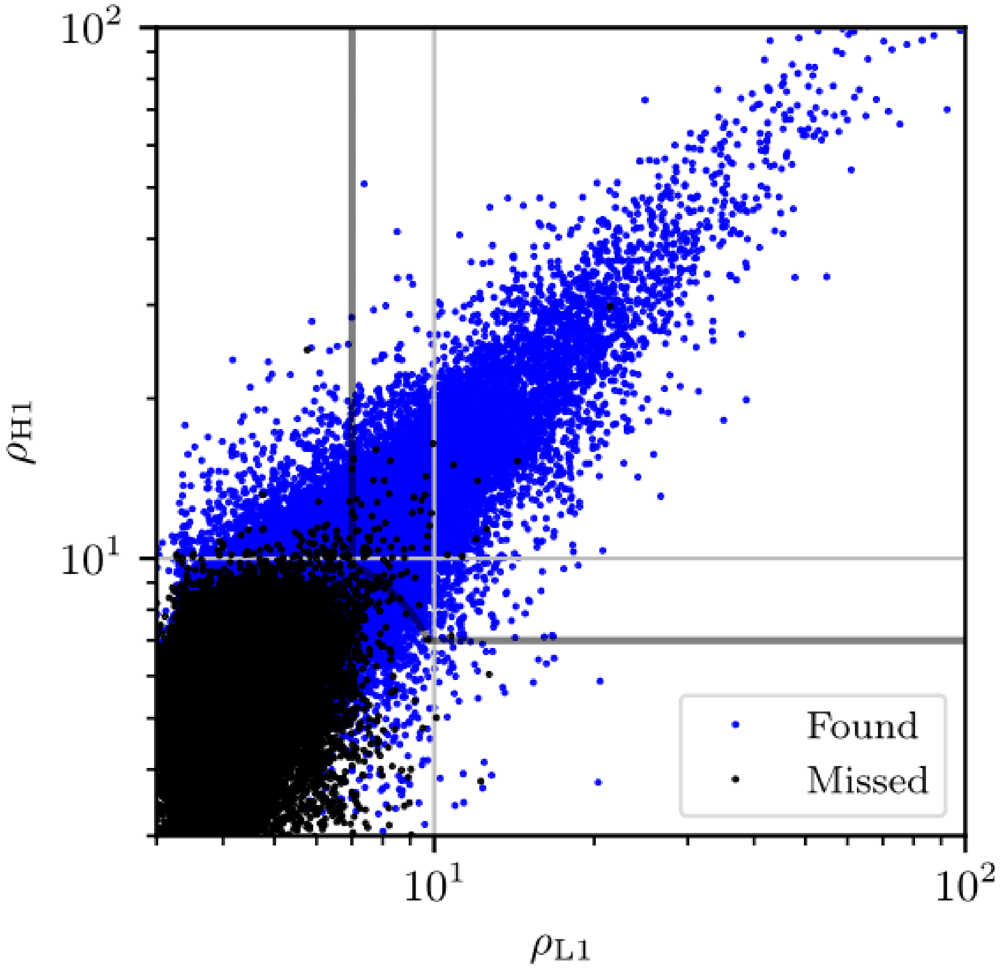


Figure 6.4. Distribution of injected SNRs for recovered injections above $\mathcal{L} = 35(\sqrt{2\mathcal{L}} = 8.37)$. Missed injections with network SNR above 12 and detector SNRs greater than 7 (indicated by the shaded contour) are discussed in the appendix.

not appear in the LIGO GWTC-2. We recover the GW170727 event with a FAR of $3 \times 10^{+1} yr^{-1}$. We recover GW170817a in Livingston with a FAR of $5 \times 10^{-3} yr^{-1}$ while Zackay reports it with a FAR of $8.7 \times 10^{-2} yr^{-1}$. Zackay also reports the probability of it being of astrophysical origin at 86% [141], but we do not make that estimation here. As in the previous case, we recover both these events to within tens of milliseconds of the previously reported values and are confident that they correspond to the respective gravitational wave candidates.

We make no claims regarding the possibility of the remaining two events we report being gravitational wave candidates. These appear eighth and ninth in Table 6.1. They are not recovered significantly, and it is likely they are noise.

The first seven events reported in Table 6.1, as well as the last one are excluded from the background, since all of them are previously reported gravitational wave candidates.

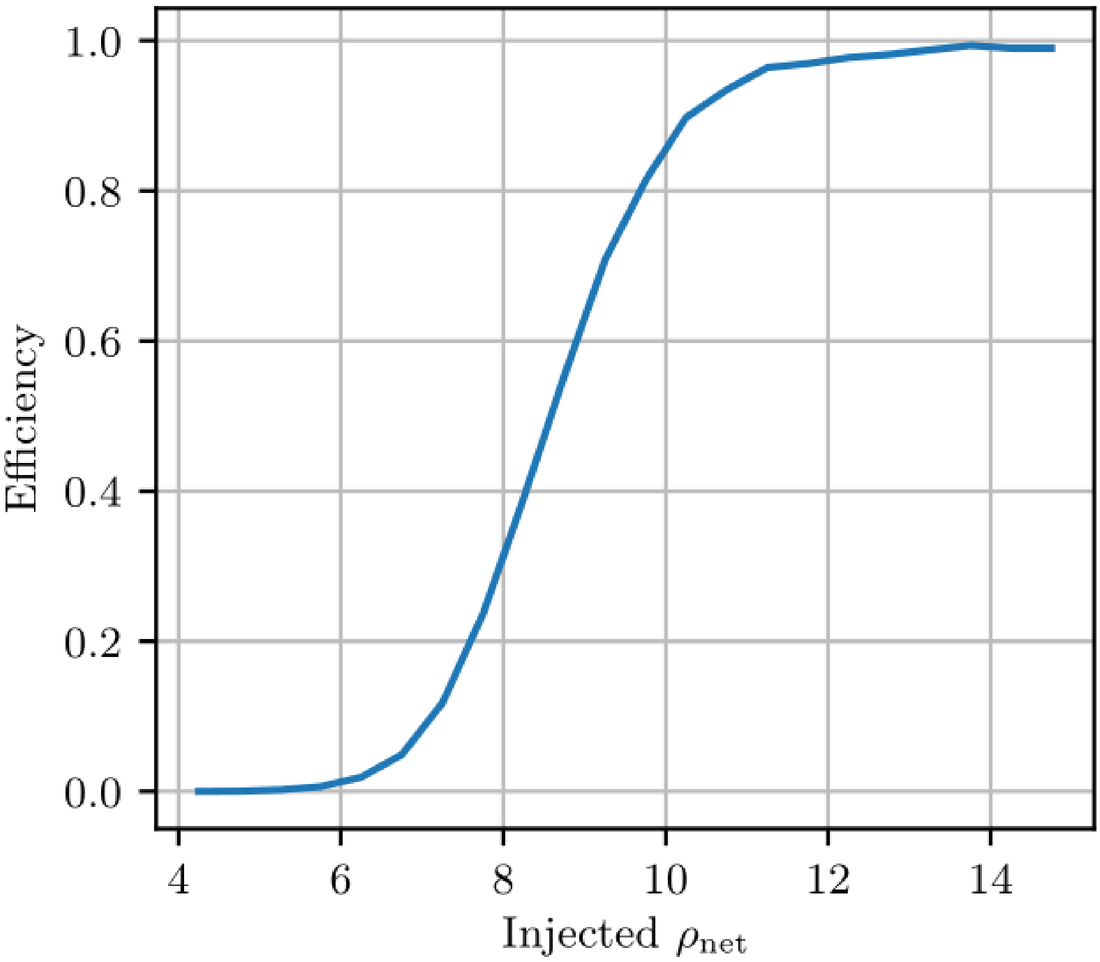


Figure 6.5. Efficiency of recovering injections at different injected SNRs

6.4.5 Sensitivity estimate

The sensitivity of our new pipeline is demonstrated in Fig. 6.4 and Fig. 6.5. Fig. 6.4 shows the distribution of all the injected events by SNR with a network SNR of 12 contour, and detector SNR of 7 contours added. This figure shows that the majority of loud injected events were recovered by our pipeline, with 70 missed in the region with network SNR above 12 and detector SNRs greater than 7. Only nine of these missed

FAR	$\sqrt{2\mathcal{L}}$	ρ_{net}	m_1	m_2	a_1	a_2	ρ_{H1}	ξ^2_{H1}	ρ_{L1}	ξ^2_{L1}	Date (UTC)
(yr^{-1})			($\mathrm{M}_\odot$)	($\mathrm{M}_\odot$)							
5×10^{-03}	18.5	18.7	1.8	1.1	-0.3	0.7	18.7	0.8	-	-	2017-08-17 12:41:04
5×10^{-03}	16.3	17.1	38.7	24.5	0.7	-0.9	9.6	1.3	14.1	1.0	2017-08-14 10:30:43
5×10^{-03}	11.9	12.6	46.4	25.8	0.6	-1.0	6.5	1.3	10.8	0.7	2017-08-09 08:28:21
5×10^{-03}	11.7	11.8	51.5	38.6	0.4	-0.5	6.6	0.9	9.8	0.7	2017-08-23 13:13:58
5×10^{-03}	10.7	10.9	73.1	43.3	-0.1	1.0	7.9	1.1	7.5	1.0	2017-07-29 18:56:29
5×10^{-03}	10.6	10.7	122.7	45.5	0.9	-0.9	-	-	10.7	1.0	2017-08-17 03:02:46
4×10^{-02}	9.6	10.1	39.7	36.3	0.7	-0.8	-	-	10.1	1.2	2017-08-18 02:25:09
$6 \times 10^{+00}$	8.7	9.0	20.8	3.3	0.1	0.6	9.0	1.0	-	-	2017-08-03 05:59:03
$3 \times 10^{+01}$	8.6	8.8	53.1	1.2	-0.3	1.0	3.7	0.9	8.0	0.8	2017-08-14 07:35:04
$3 \times 10^{+01}$	8.5	8.7	51.5	43.6	0.4	-0.9	4.6	0.8	7.4	1.1	2017-07-27 01:04:30

Table 6.1. Candidate gravitational wave events with the 10 smallest false alarm rates and largest SNRs. The first seven triggers as well as the last one correspond to known gravitational wave candidates: GW170817, GW170814, GW170809, GW170823, GW170729, GW170817a, GW170818, and GW170727. The other two triggers have not been previously reported as gravitational wave candidates.

injections are because the pipeline could not adequately recover the injections. This shows that the pipeline only very rarely gets stuck at local peaks, instead of finding the global maxima, which will correspond to the injected signal. It is possible that as we move to a lower mass parameter space, the frequecy of such occurrences will increase. All of the loud missed events are discussed in more detail in Appendix B.

Fig. 6.5 shows the efficiency of the pipeline as a function of the injected network snr of the synthetic gravitational wave set described in section C. This plot shows that without any data cleaning implementation, almost 90% of events at SNR 10 are recovered by the pipeline while that percentage only increases with the SNR and plateaus just short of 100% around SNR 13.

6.5 Conclusion

In this paper, we have described in detail a novel gravitational wave detection algorithm. This algorithm searches stochastically over the chosen parameter space, saving the time and computing power required to generate large banks of template waveforms. The algorithm samples the parameter space by making jumps with a pre-estimated mismatch between templates informed by the parameter space metric and keeping those points which have a higher SNR. This method is shown to be of comparable accuracy in the recovery of gravitational wave events at high masses as current template-based pipelines.

To demonstrate the validity of this method, we have presented an analysis of approximately one month of LIGO data from July 27 00:00 2017 UTC – Aug 25 22:00 2017 UTC exploring the binary black hole parameter space. We recovered six known gravitational wave candidate events to within tens of milliseconds of previously reported coalescence times, as well as two gravitational wave candidates previously reported.

Additionally, we conducted an injection campagin of compact binary mergers to prove the sensitivity of the pipeline to binary black hole merger events. We recovered almost 90% of events with SNR 10 and an increasing percentage at higher SNRs that plateaus just below 100% at SNR 13. The majority of the missing loud injections were due to the presence of glitches near the injected events.

In the future, we plan to extend our method to all regions of the parameter space. We expect that even though the algorithm will scale similarly to any search using template banks at lower mass, it will still retain its other advantages, such as simpler workflow and ease of setup. We plan to make our method competitive with other searches like GstLAL for LIGO's fourth observing run. It remains an open project to get good convergence during the sampling process for all regions of the parameter space.

6.6 Acknowledgements

This research has made use of data, software and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org/), a service of LIGO Laboratory, the LIGO Scientific Collaboration and the Virgo Collaboration. LIGO Laboratory and Advanced LIGO are funded by the United States National Science Foundation (NSF) as well as the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany for support of the construction of Advanced LIGO and construction and operation of the GEO600 detector. Additional support for Advanced LIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain.

This work was supported by National Science Foundation awards OAC-1841480, PHY-2011865, and OAC-2103662. Computations for this research were performed on the Pennsylvania State University's Institute for Computational and Data Sciences gravitational-wave cluster. CH Acknowledges generous support from the Eberly College of Science, the Department of Physics, the Institute for Gravitation and the Cosmos, the Institute for Computational and Data Sciences, and the Freed Early Career Professorship.

6.7 Appendix A: Data release details and code versions

A tarball containing the source code and data files necessary to reproduce the results and plots in this paper can be found at https://dcc.ligo.org/T2100321. Instructions for installing the code and for using it to create the plots and results can be found in README.md inside the source_code directory in the tarball.

6.8 Appendix B: Follow-up of missed injections

In this appendix, we will discuss the particularly loud injections which were not recovered during the simulation mode of the pipeline. An injection is deemed to be recovered, if it

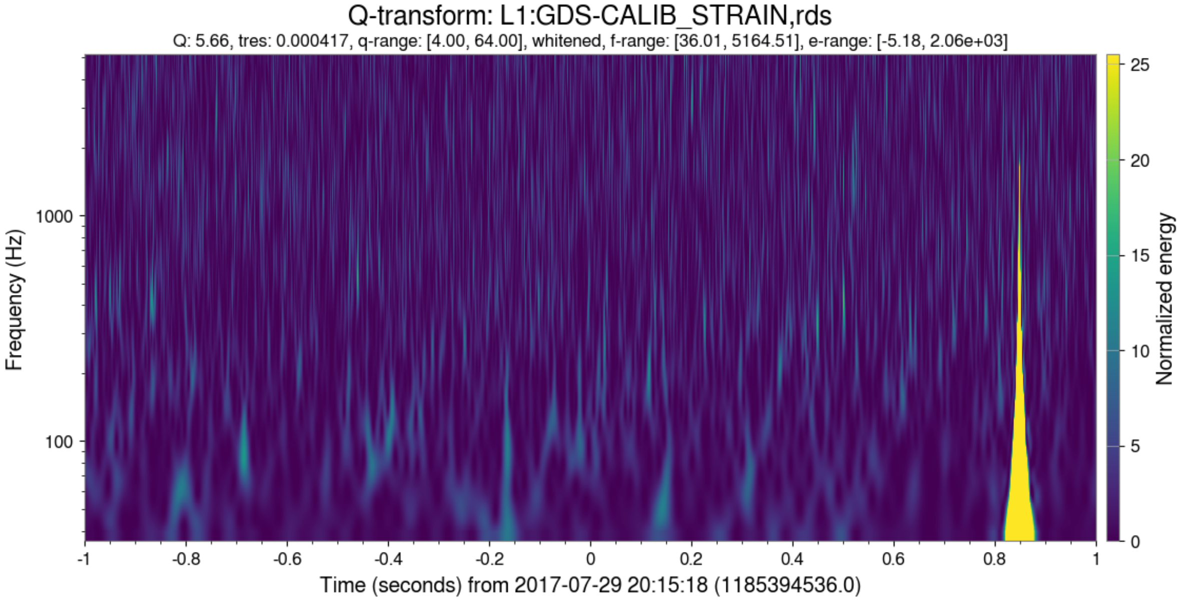


Figure 6.6. An example of a Q-transform plot showing a glitch in Livingston, causing a simultaneous injection to be missed

was assigned a log-likelihood ratio, $\mathcal{L}$ of 35 or greater. Out of the 112526 injections, 65361 were missed. Most of these (65291 out of 65361) were missed because the injected SNR was too low for them to be recovered significantly. Some, however had a high injected SNR and were still missed. We will discuss the reasons for the same, for missed injections with network SNR above 12 and detector SNRs greater than 7. These injections are shown in Fig. 6.4, of which there are 70. Out of these, 33 were missed due to the data containing a glitch simultaneous to the injection, causing the glitch rejection mechanism to reject that part of the data. The existence of a glitch in the data was verified by creating Q-transform plots of the data window. An example of such a glitch is shown in Fig. 6.6. Out of the remaining 37 loud missed injections, 28 fell into the glitch region as defined in Sec. 6.3.4.6, and hence were rejected. The pipeline failed to recover only 9 injections out of the original 112526. However, such problematic injections can be recovered by increasing N_k, the number of samples to reject at each level, k, before moving on to the next, at the cost of the run-time of the pipeline.

Chapter 7
Method to get Better Sky Maps in a GstLAL Low-Latency Analysis

This is a reprint of [73].

Modeled gravitational wave searches correlate the strain data with a bank of gravitational waves template waveforms to make detections of gravitational wave candidates, and these results are processed by downstream tools to calculate the likely sky location and distance of the source of the candidates. This is crucial for multi-messenger efforts, since it informs astronomers where to point their telescopes to facilitate electormagnetic follow-up of the gravitational wave candidates. We present a novel method to improve the low-latency results of the GstLAL gravitational wave seach pipeline, and thus improving sky location estimates of low-latency candidates. This method involves ingesting the GstLAL low-latency results, and performing a small targeted hierarchical search to recover the candidates with more accurate parameters, in a medium-latency timescale (few seconds to five minutes). To test our method, we perform a GstLAL low-latency analysis on forty days of data from the third observing run of LIGO, Virgo, and KAGRA, and show that our method improves the GstLAL results by 5.38% and the subsequent sky location results by 16.75% on average. In addition to this increase in precision, we also show that these results are more accurate as compared to the GstLAL results. This method has been adopted by GstLAL for the fourth observing run.

7.1 Introduction

Since the second observing run (O2) of the LIGO Scientific, Virgo, and KAGRA (LVK) collaboration, gravitational waves (GWs) have emerged as an important messenger in multi-messenger astronomy. It was during this observing run that GW170817 was detected [14, 36, 37]. The source of this event was a binary neutron star (BNS), and thus electromagnetically bright, leading to the first multi-messenger detection involving gravitational waves (GWs). It has since led to a variety of new scientific results [38, 39].

Participation of GWs in multi-messenger detections like GW170817 is made possible by a combination of multiple analysis pipelines and tools working together. First, a GW search pipeline analyzes the strain data produced by GW detectors like the Laser Interferometer Gravitational-Wave Observatory (LIGO) Hanford and Livingston detectors [11], the Virgo detector [12], and the KAGRA detector [13], finds GW candidates in the data, and uploads them to the Gravitational Wave Candidate Event Database (GraceDB) [35] in near-real time. Examples of GW search pipelines are GstLAL [19–22], IAS [33, 34], MBTA [29, 30], PyCBC [26–28], and SPIIR [31, 32].

Next, GW inference pipelines ingest the results of GW searches, and infer source properties like the source parameters, sky location, distance, etc. Sky location is generally communicated in the form of a sky map, a two-dimensional plot of the sky showing contours for probable location of the source. Two commonly used contour values are 50% and 90%. Examples of sky maps are shown in Fig. 7.1. LALInference [164, 165], BAYESTAR [45, 46], and BILBY [47, 48] are some of the inference pipelines used.

After the sky map for a GW candidate is calculated, a public alert gets sent [49]. All of this happens within seconds to minutes of the GW signal reaching Earth. Astronomers can then choose to point their telescopes at the location described by the sky map included in the public alert, to try to observe any electromagnetic counterparts to the GW candidate. However, these electromagnetic counterparts can be very faint and can fade within seconds of the GW detection. Because of this, it is crucial for multi-messenger efforts to extract all possible information from the GW signal to give astronomers the best opportunity to observe any electromagnetic counterparts.

How well constrained the source is in a sky map depends very heavily on the number of detectors contributing to the detection of the candidate. A single detector gives almost no information. Two detectors generally localize the source to a circle in the sky. Three detectors, however, can localize the source to a few points in the sky. For example, in Fig. 7.1, the left panel is derived from a candidate comprising three detectors, whereas

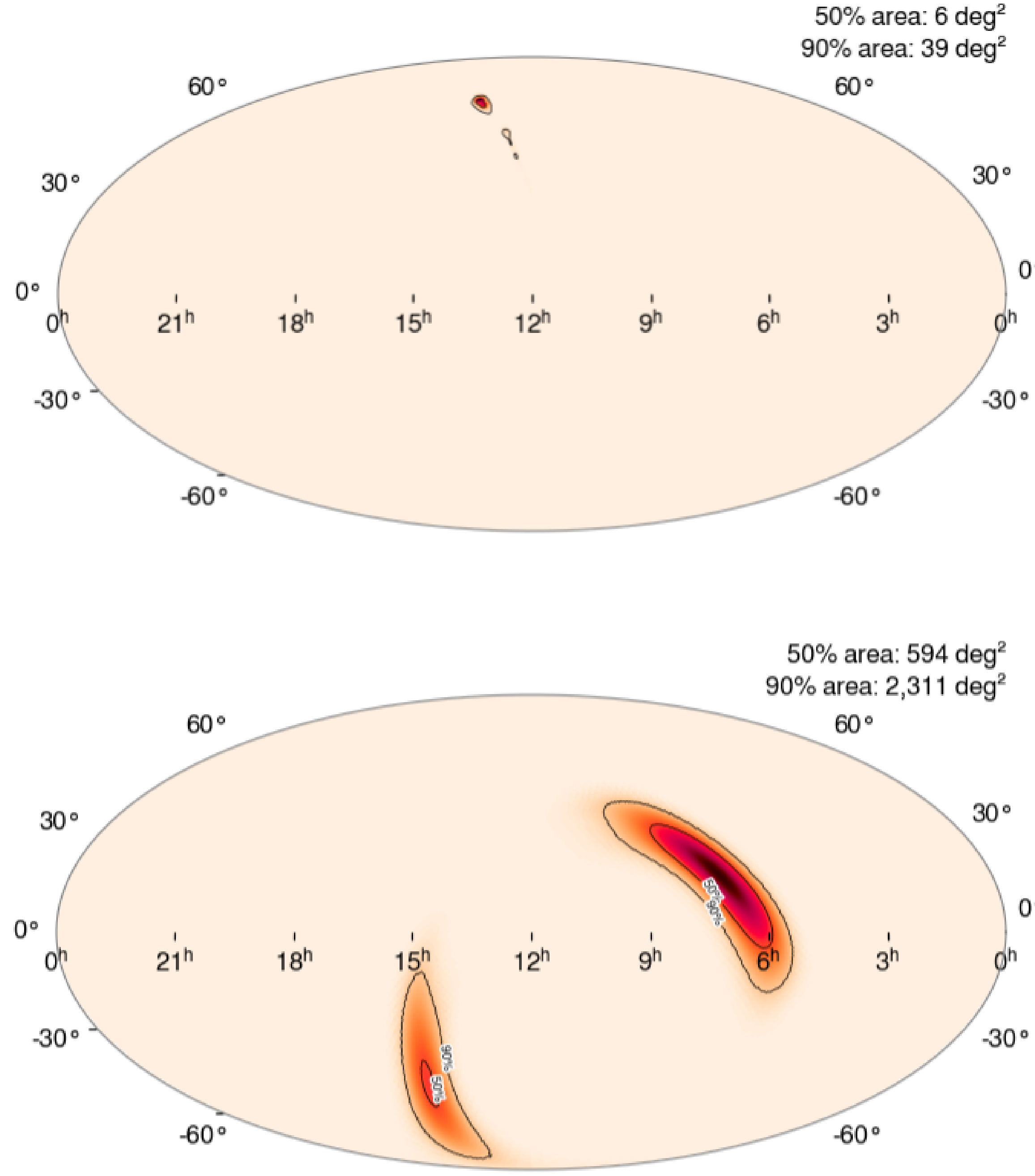


Figure 7.1. This figure shows two examples of sky maps. The one on the top is relatively well constrained in terms of sky location, while the one of the bottom is less constrained. Louder GW signals being detected in more number of detectors produces more constrained, and hence better sky maps. These sky maps were produced using the BAYESTAR package on simulated GW signals.

the right panel is derived from a candidate comprising two. With more than three detectors, even smaller points can be obtained. Consequently, efforts are constantly taken to minimize detector downtime, and to coordinate maintenance schedules among among detecotrs to maximize coincident observing time.

Another way to obtain better sky maps is to construct better algorithms to extract more information from a GW signal. However, that discussion is out of scope for this work. Instead, we will focus on the final way to improve sky maps, which is to improve the results produced by GW searches, from which the sky location information gets extracted by inference pipelines. They do this by looking at the arrival times, phases, and amplitudes of the GW signal at the different detectors, as reported by the GW search. Consequently, by measuring these values more accurately, GW searches can contribute to better sky maps.

In this work, we introduce the GstLAL SNR Optimizer, which is designed to do just that. GstLAL, like other modeled GW searches uses a collection of GW template waveforms, each with a different combination of source parameters, called a “template bank”, and correlates each one of them across the GW strain data, in a process called matched filtering. The output of this process is called the signal-to-noise ratio (SNR) time series. However, since the parameter space of GW sources is only discretely sampled by the template bank, we expect a loss in the SNR, and in the accuracy of the arrival times and phases measured by GstLAL. Common values of the minimal match between neighboring templates in a template bank range from 0.97 to 0.99 [41]. As a result, we expect around a 1 to 3% loss in the SNR. The GstLAL SNR Optimizer, referred to as the SNR Optimizer hereafter, ingests GstLAL low latency results in real time, and performs a small targeted hierarchical search. The search is targeted in two ways: The SNR Optimizer only analyzes the data close in time to a candiate reported by GstLAL, and it also analyzes the parameter space close to the parameters reported by GstLAL. It does not analyze a fixed bank of templates, but rather dynamically creates new ones in the relevant parameter space, hierarchically closing in on the true location of the signal in the parameter space. It also implements other improvements, like more accurate matched filtering, leading to higher SNRs, and hence better sky maps. It does all this in a timescale of a few seconds to five minutes, and if it manages to produce better results, the skymap produced from its results is included in the public alert issues for the candidate.

In Sec. 7.2, we will discuss the detailed implementation of the SNR Optimizer, and in Sec. 7.3, we will discuss the tests that we ran to measure the performance of the SNR

Optimizer.

7.2 Methodology

7.2.1 General GstLAL Methods

As described before, modeled GW searches like GstLAL make use of a template bank that discretely samples the intrinsic parameter space (i.e. masses and spins) of the GW sources, causing a loss in SNR. Before the data can be matched filtered with the templates, both the data and the templates are whitened using a power spectral density (PSD) that represents the frequency characteristics of detector noise. GstLAL implements whitening in the frequency domain, and matched filtering in the time domain. The whitened data is defined as

$$\hat{d}(\tau) = \int_{-\infty}^{\infty} df \frac{\tilde{d}(f)}{\sqrt{S_n(|f|)/2}} e^{2\pi i f \tau} \tag{7.1}$$

where $\tilde{d}(f)$ is the data represented in the frequency space, and $S_n(|f|)$ is the single-sided PSD. Similarly, the whitened template is defined as

$$\hat{h}(\tau) = \int_{-\infty}^{\infty} df \frac{\tilde{h}(f)}{\sqrt{S_n(|f|)/2}} e^{2\pi i f \tau} \tag{7.2}$$

where $\tilde{h}(f)$ is the template represented in the frequency space. The matched filter output for that particular template, i.e. the SNR is then calculated as

$$\mathrm{SNR}(t) = \int_{-\infty}^{\infty} d\tau \hat{d}(t+\tau)\hat{h}(\tau) \tag{7.3}$$

GstLAL currently has the ability to whiten the data in real time, but template whitening needs to be done before starting the process of matched filtering. For a low-latency analysis, since the data are not known beforehand, this means that the templates are whitened using a PSD projected to represent future detector noise [4]. This mismatch between the PSD used to whiten the templates, and the "true" PSD measured from detector noise also causes a loss in SNR.

Since ensuring the least possible latencies for public alerts is crucial for multi-messenger astronomy, the GstLAL low-latency analysis makes certain concessions on the quality of matched filtering. These include using a relatively short length of data (i.e. an FFT length of 4 seconds of data) for measuring the PSD and whitening the data in Eq. (7.1). This results in lower statistics while measuring the PSD, and produces a PSD sampled at relatively larger intervals of frequency. Similarly, the data itself is sampled at a relatively lower rate of 2048 Hz. Both of these concessions case a decrease in the SNR.

The GstLAL analysis recognizes times during which the SNR of some template crosses the threshold value of 4, called a “trigger”. Triggers recognized as originating from noise [42] are added to the background. Triggers are then ranked against this background, and a likelihood ratio (LR) [23] is calculated as a ranking statistic. Triggers with high LRs are called GW candidates. The LR is then converted to a false alarm rate (FAR) by taking into account the LR statistics of noise triggers, as well the livetime of the analysis. For a low-latency analysis, if the FAR crosses a specific threshold [44], the candidate is uploaded to GraceDB, and a public alert is issued [49]. The public alert contains information about the sky location of the source, low-latency parameter estimation of the source [50], as well as information about the probability of astrophysical origin for different source classes [51–53]. All of this can help astronomers follow-up on any potential electromagnetic counterparts. A detailed description of the GstLAL low-latency analysis can be found in [43].

In summary, a GstLAL low-latency analysis loses SNR due to the following reasons:

1. discrete nature of the fixed template bank
2. templates not whitened with a PSD measured in real time
3. low FFT length for PSD measurement and data whitening
4. low rate of data sampling

7.2.2 SNR Optimizer Methods

7.2.2.1 Design Principles

The SNR Optimizer is designed to mitigate any loss in SNR due to the aforementioned reasons, and maximize SNR, leading to better sky maps. It implements a targeted, hierarchical, and sub-threshold search in order to follow-up on candidates reported by GstLAL, and recover them with a higher SNR. The SNR Optimizer has evolved from the metric assisted stochastic sampling (MASS) GW search described in [137]. As such, the SNR calculation for any given template is identical to that performed by MASS. A central philosophy of the SNR Optimizer is that it does not attempt to evaluate the significance of a candidate. Instead it relies on the GstLAL low-latency search not only to provide it with candidates that arise from astrophysical sources rather than noise, but also to provide the LR, FAR, and the probability of astrophysical origin for different source classes for these candidates.

7.2.2.2 Template Bank

The SNR Optimizer does not rely on a fixed bank of pre-created templates. Instead, because of its hierarchical nature, it starts off with a fixed template bank, and dynamically creates new templates in between the gaps of the original templates, as it closes in on the true location of the signal in the intrinsic parameter space. To do this, it makes use of template banks created by the `manifold` [40] software package.

`Manifold` implements a metric on the intrinsic parameter space in order to place templates at a fixed mismatch from each other, and this metric information at the location of every template is stored in the template bank file. Templates are expressed as hyper-rectangles in the intrinsic parameter space. Each such rectangle comprises a fixed area in the intrinsic parameter space (depending on the dimensionality of the space, this might actually be a 3D or 4D volume), the template at the center of this area, and the metric at the location of the template Using the metric, any template can be split into two new templates, each occupying half the area of the original template. The metric gets re-calculated at the locations of the new templates enabling this process to be repeated indefinitely. This operation is cheap, since the calculation is done only approximately, enabling information from the metric at the original template to be re-used. This process is illustrated in Fig. 7.2. This easy splitting of templates into multiple new ones enabled by `manifold` facilitates the hierarchical nature of the SNR Optimizer.

7.2.2.3 Search Algorithm

Similar to GstLAL, the SNR Optimizer can be run in a low-latency “online” mode, or a high-latency “offline mode”. In the offline mode, it reads in a gstlal event file for a particular candidate, and analyzes a small amount of data around it. The offline mode is meant only for testing purposes. More importantly, in the online mode, the SNR Optimizer continuously reads in live data, continuously calculates the PSD from the data, and whitens it with that PSD. Because it operates in a medium-latency timescale, it can afford to use a higher FFT length than GstLAL for the purposes of data whitening. It can also sample the data at higher rate than GstLAL, since it does not matched filter all of the data, but rather only short amounts of data around candidates, and hence there

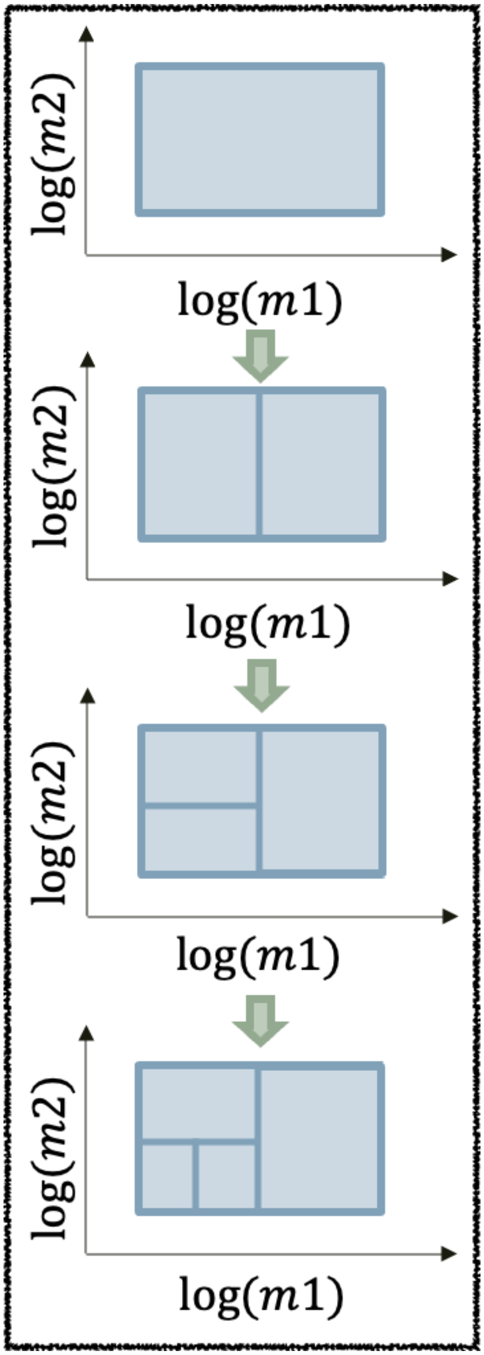


Figure 7.2. This figure shows a schematic of the template splitting process enabled by the manifold rectangle structure. Each rectangle has a template at its center, and the metric is calculated at that location. The boundaries of the rectangle represent the area (or volume, depending on the dimensionality of the space) occupied by the template. This schematic shows a template splitting into two new templates, one of which again splits into two new templates, and again one more time. This schematic shows templates only in the m_1-m_2 space. For the fourth observing run, the SNR Optimizer operates in the m_1-m_2-χ_{eff} space.

is no danger of falling behind live data. Both of these factors contribute to the higher SNRs obtained by the SNR optimizer.

Similar to GstLAL, the SNR Optimizer also implements “gating” on the whitened data, in which if the amplitude of whitened data exceeds a certain number of standard deviations, it is instead set to 0, with 0.25 seconds of padding on either side [19, 21]. This is done to remove any transient non-gaussian components in the noise, called “glitches”, which can artificially increase SNR and mimic GWs.

While it is doing this, it also listens for any candidates that the GstLAL low-latency analysis reports. When it receives such a candidate, it performs matched filtering on the data in a 2 second window (plus sufficient padding on either side) around the event time as reported by GstLAL. Because it was continuously whitening data before, data whitening does not need to be done from scratch for the candidate, helping with latencies. It does not filter the data using all the templates in the bank, but rather chooses the 1000 closest templtes to the template reported by GstLAL. Because the metric at each template it known, the operation for finding the closest templates to a given template is a cheap one. The underlying assumption in only choosing 1000 templates to filter is in accordance with the central philosophy of the SNR Optimizer, that GstLAL will only provide astrophysical candidates to the SNR Optimizer, and as such the template reported by GstLAL is close to the “true” template. All templates are whitened in real time, using the latest estimate of the PSD. In this way, the SNR Optimizer sets up a targeted search around the candidate.

After it performs matched filtering for those 1000 templates, triggers are formed from all 1000 templates using SNRs from all detectors. Out of those, only 500 templates with the highest network SNR are retained, while the rest are discarded. These 500 templates are then each split into 2 new templates, and the SNR at each of these new templates is calculated, and triggers are formed. This cycle is repeated, now using the 500 retained templates, as well as their child templates, and this time around, only 250 templates get retained.

There are 2 stopping conditions to this loop. A template will not split further if its area is lower than some threshold value. This would mean that further splitting it would only give templates very close by, cauing diminishing returns. If at any iteration of the loop, no new templates can be formed by this process of splitting, that means we’ve arrived at our final set of templates, and the template whose trigger has the highest network SNR is chosen as the temmplate and the candidate reported by the SNR Optimizer. The second stopping condition is if the SNR Optimizer reaches 4 minutes of processing time from the time it received the candidate from GstLAL. When that happens, the SNR Optimizer takes whatever set of templates it had retained up to that point, and reports the template with the highest network SNR trigger as its template and candidate. The reason for this timeout is that the LIGO Scientific, Virgo and KAGRA Collaboration (LVK) has an internal timout of 4.5 minutes [166] for candidates to be considered for generating a sky map and issuing a public alert. In this way, the SNR Optimizer implements a hierarchical search to recover the candidate with a template very close to the true template of the GW signal.

7.2.2.4 Coincidence Formation

The method implemented by the SNR Optimizer for forming coincidences across detectors when creating a cndidate is different than that implemented by GstLAL, because of two reasons:

1. The SNR Optimizer is a sub-threshold search, whereas GstLAL is not
2. Since GstLAL calculates a LR based on the properties of the coincidence itself, it can afford to be less strict when forming coincidences, because if it forms a "bad" coincidence (i.e. with an unphysical cobination of arrival times, phases and SNRs at different detectors), the part of the LR equation that calculates the probability of such a combination of times, phases and SNRs, commonly called the $dt - d\phi - dSNR$ term of the LR, will downrank this trigger, and it will not become a candidate. However, since the SNR Optimizer outsources its LR calculation to GstLAL, it needs to make sure the coincidence it forms is physical

In order to create coincidences in a sub-threshold search, the SNR Optimizer first finds peaks in the SNR timeseries for every detector independently, in the 2 second window of data being analyzed. It then loops over these peaks, and for every one, it adds the other detectors by finding new peaks in such a way that the combination of arrival times at the detectors remains physical. No arrival phase or SNR information is used to assess the physicality of the coincidence, and in the next paragraph, we will show that this is enough to guarantee physical coincidences. Whenever new detectors are being added, the order of addition is kept the same as the descending order of SNRs of the original SNR peaks. This is done because we want to add louder detectors earlier, when there are less constraints on them for keeping the coincidence physical. After the loop over all detectors is finished we are left with coincidences equal in number to the number of detectors. The coincidence with the maximum network SNR is then chosen to be the candidate reported by the SNR Optimizer for that window of data.

The two LIGO detectors and the Virgo detector form a plane, and hence the arrival times of a candidate at the detectors can be converted into information of the arrival direction of the GW and its velocity parallel to this plane, by solving the following system of equations:

$$t_1 = \frac{\mathbf{n} \cdot \mathbf{r_1}}{c} \tag{7.4}$$

$$t_2 = \frac{\mathbf{n} \cdot \mathbf{r_2}}{c} \tag{7.5}$$

$$t_3 = \frac{\mathbf{n} \cdot \mathbf{r_3}}{c} \tag{7.6}$$

where t_i are the arrival times at the detectors, $\mathbf{n}$ is the vector representing the direction of the GW, and r_i are the location vectors of the three detectors. If the velocity of the GW parallel to the plane so obtained is less than or equal to c, the speed of light, the combination of times are phyical. This argument can be condensed into the calculation of a χ^2 statistic:

$$\chi^2 = (t_1 - \frac{\mathbf{n}_\parallel \cdot \mathbf{r_1}}{c} - t)^2 + (t_2 - \frac{\mathbf{n}_\parallel \cdot \mathbf{r_2}}{c} - t)^2 + (t_3 - \frac{\mathbf{n}_\parallel \cdot \mathbf{r_3}}{c} - t)^2 \tag{7.7}$$

where t is the average arrival time that minimizes the expression. For a physical combination of times, χ^2 will be 0, whereas for an unphysical combination, it will be larger than 0. While forming coincidences, only those arrival times that give a χ^2 less than 3 are used. When adding a second detector to a coincidence, this just means selecting the highest SNR peak within the light travel time of the two detectors. However, when adding the third detector, all possible sample points within the light travel time of the two earlier detectors are considered, and the maximum SNR one that gives a χ^2 less than 3 is added. The reason for selecting a χ^2 threshold of 3 instead of 0 is that due to numerical noise, even a physical combination of time will not give a χ^2 perfectly equal to 0, but rather a very small value. A second reason is that due to detector noise, the arrival time estimate might not be perfect, and the χ^2 threshold needs to allow for that.

Since this only considers information from the arrival times, and not phases or SNRs, this is necessary, but not sufficient for forming a physical coincidence. However, we will show that considering time information is also sufficient. Fig. 7.3 shows that $dt - d\phi - dSNR$ contains more information than χ^2, but despite that the two are well correlated. If we had used a threhold on $dt - d\phi - dSNR$ to qualify a trigger as physical

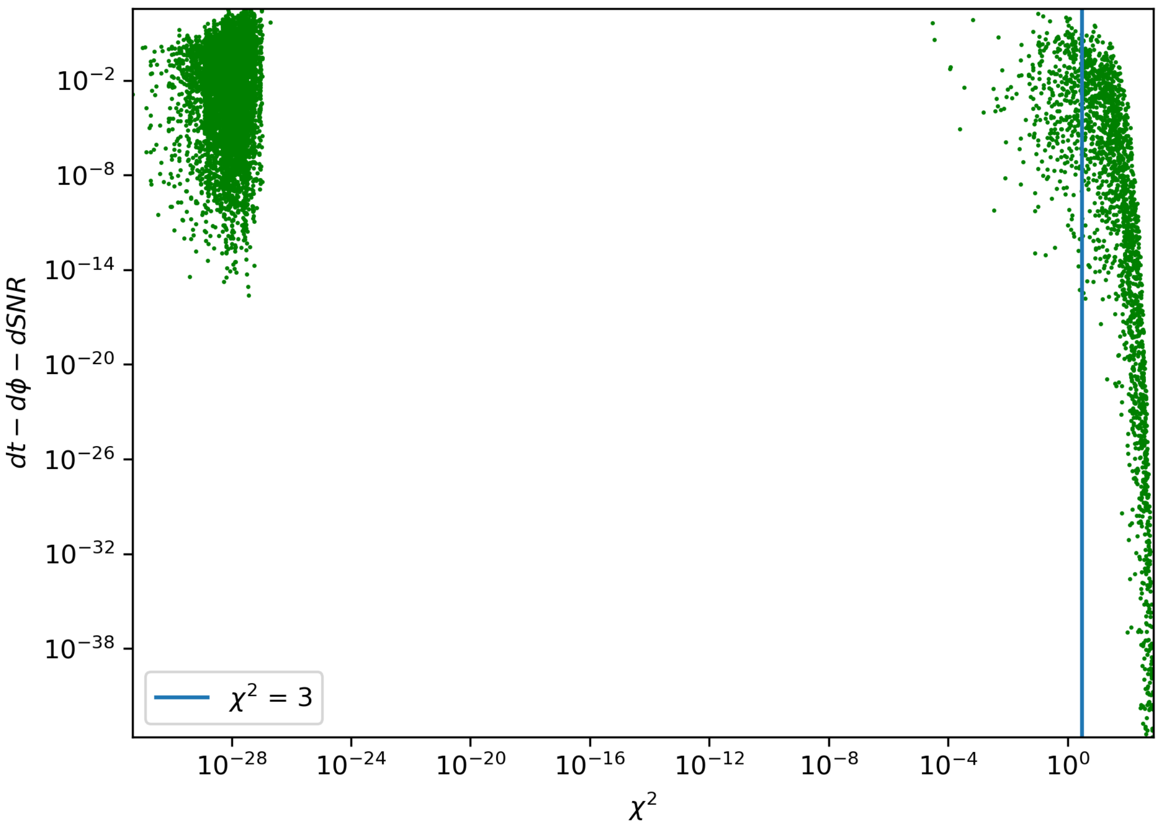


Figure 7.3. This figure shows the χ^2 value plotted against the $dt - d\phi - dSNR$ term of the LR implemented by GstLAL, for various triggers. The two quantities, while not perfectly correlated, are well correlated, and selecting the $\chi^2 = 3$ threshold for calling a coincidence physical is almost equivalent to selecting a threshold value for $dt - d\phi - dSNR$. The tiny values of χ^2 seen on the left side of the plot are numerical noise in the calculation of χ^2, and actually represent a value of 0.

instead, it would have had a very similar effect to the $\chi^2 = 3$ threshold. In Fig. 7.4, we see that the $\chi^2 = 3$ threshold is able to perfectly distinguish between simulated gravitational wave triggers (and hence guaranteed to be physical) and triggers generated by randomly drawing arrival times (and hence unlikely to be physical triggers).

7.2.2.5 O4 Configuration

The SNR Optimizer has been adopted by GstLAL for use during fourth observing run (O4). Here, we will provide details about the configuration used for the SNR Optimizer in O4. The initial template bank used by the SNR Optimizer is the full O4 GstLAL template bank, since it has been generated via `manifold`. It is generated in

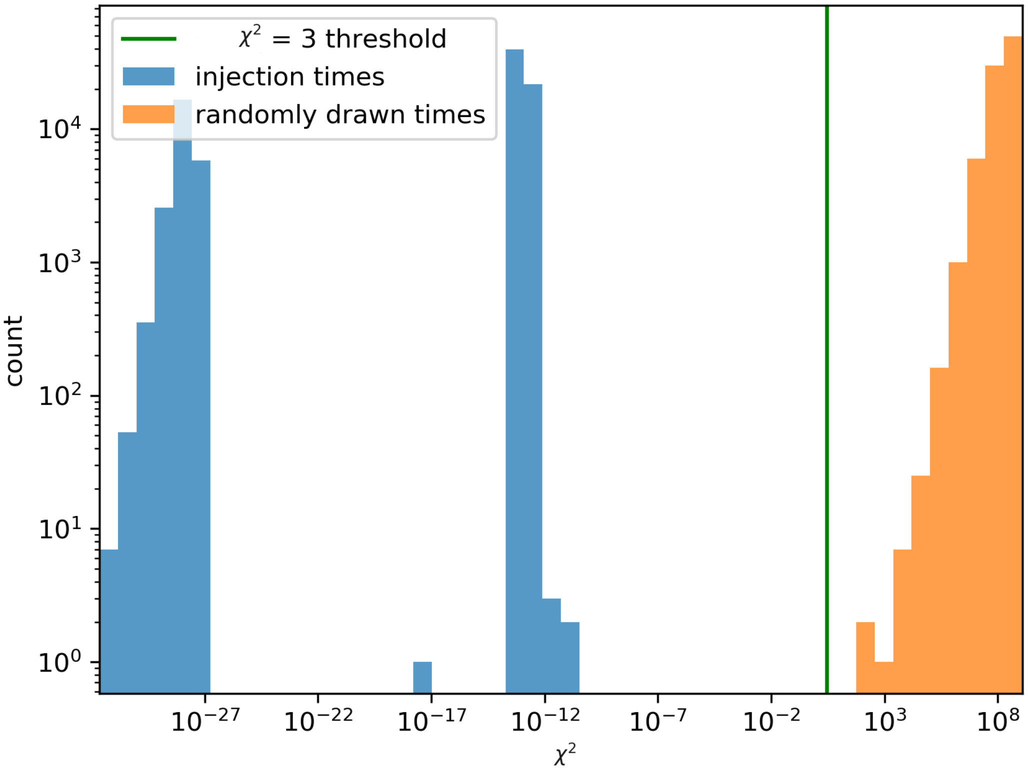


Figure 7.4. This plot shows a histogram of the χ^2 value for two types of triggers. The first, represented in blue, are simulated GW signals, and are guaranteed to be physical. The second, represented in orange, are triggers with randomly drawn times, and are highly unlikely to be physical. We see that the $\chi^2 = 3$ threshold serves as a perfect discriminator between physical and unphysical triggers. The tiny values of χ^2 for the injection triggers are numerical noise in the calculation of χ^2, and actually represent a value of 0.

the 3 dimensional parameter space of m1-m2-χ_{eff}, where χ_{eff} is the component mass weighted average of the dimensionless spins. It contains around 2 million templates with component masses from 1-200$M_\odot$ and dimensionless spins up to ±0.99. More information about the O4 GstLAL template bank can be found in [41].

For matched filtering, the FFT length for PSD measurement and data whitening is set to 16 seconds worth of data, and the rate of sampling the data is set to 8192 Hz. Both of these values are larger than those used by GstLAL, which uses an FFT length of 4 seconds of data, and a sample rate of 2048 Hz. The threshold of standard deviations after which to gate the whitened data is set to 15. The stopping criterion of the minimum area of a rectangle below which it will not split into new templates is set to the area corresponding to a mismatch of 0.001.

Communication between the GstLAL low-latency analysis and the SNR Optimizer is handled via the `Apache Kafka` software [167]. When a GstLAL low-latency job finds a candidate, it sends a `kafka` message to the "events" `kafka` topic. From here, a specialized uploader job reads this message and uploads candidates to GraceDB. Since there can be multiple triggers reported by different GstLAL jobs for the same candidate, the uploader job checks at regular intervals and only uploads the best trigger for a given candidate. The best trigger is defined as the trigger with the lowest FAR, if the FAR of the trigger is greater than the public alert threshold [44], and the trigger with the highest SNR otherwise. After it uploads a trigger, it sends a `kafka` message to the "uploads" `kafka` topic. To account for multiple uploads in a short burst of time, 10 SNR Optimizer jobs are running in parallel. Due to the partition structure of `kafka` topics, the message to the uploads topic automatically gets assigned to one particular SNR Optimizer job. After that job finishes its processing and finds a trigger for the candidate, it sends a `kafka` message to the uploads topic, from where the uploader job can upload this trigger if it has a higher SNR. If it does so, SNR Optimizer jobs will not re-trigger on the message the uploader job will send to the uploads topic.

7.3 Results

7.3.1 Data set

In order to test the performance of the SNR Optimizer, we set up a GstLAL low-latency analysis, along with the SNR Optimizer, on data from third observing run (O3). This was part of a Mock Data Challenge (MDC), and the O3 data was streamed from Jan 02 18:39:42 UTC 2024 to Feb 11 18:39:42 UTC 2024. The MDC also included an injection campaign. Injections are simulated gravitational wave signals added to the data, and their purpose is to test the performance of the analysis. More information about the MDC and the injection distribution used can be found in [5].

During this time, the GstLAL low-latency analysis made 14710 uploads from the injection part of the analysis. For these GstLAL candidates, the SNR Optimizer was able to find a higher SNR for 10259 of them, representing 70% of the total GstLAL uploads. Since GstLAL uploads multiple triggers for the same candidate, after accounting for this fact, there were 5022 distinct injection candidates that GstLAL had uploads for. Out of these, the SNR Optimizer had the highest SNR for 2940 of them. The trigger having the highest SNR for a candidate if called the "preferred event". In other words, the SNR Optimizer was the preferred event for around 60% of candidates found by GstLAL. These injection uploads from the GstLAL low-latency analysis as well as from the SNR Optimizer will be used for calculating the results presented below.

7.3.2 SNR improvement

By comparing the network SNRs of the GstLAL uploads and the corresponding SNR Optimizer uploads, we can create a histogram of combined SNRs improvement due to all methods implemented by the SNR Optimizer. This histogram is shown in Fig. 7.5. We see that on average, there's a 5.38% improvement in SNR. A point to note here is that since the job responsible for uploading both the GstLAL and SNR Optimizer triggers to GraceDB only does so if the trigger has a higher SNR than previous triggers uploaded for the same candidate, only the 70% cases in which the SNR Optimizer found a higher SNR than GstLAL participate in this histogram.

Additionally, we can plot the mean SNR improvement for different SNR and inverse FAR thresholds. This is shown in Fig. 7.6. It shows that the SNR Optimizer is most effective at lower SNRs or inverse FARs, and the effectiveness goes down slightly with an increase in SNRs or inverse FARs. Similarly, we can also plot the average percent of times the SNR Optimizer is the preferred event for a candidate for different SNR and inverse FAR thresholds. This is shown in Fig. 7.7. Similar to the previous figure, we see the same trend of the effectiveness of the SNR Optimizer going down slightly with an increase in SNRs or inverse FARs.

7.3.3 Sky map improvement

In this section, we will calculate the improvement in the sky map produced by the higher SNR of the SNR Optimizer triggers, as compared to GstLAL triggers. The following tests are usually used to evaluate the performance of an inference pipeline by assuming the input trigger files are correct (see [45, 168]), but for our purposes, we will use it to evaluate the performance of two sets of trigger files, one from the SNR Optimizer and one from GstLAL, assuming the inference pipeline is correct.

In order to create a sky map from the trigger files of the SNR Optimizer and GstLAL, we use the BAYESTAR [45, 46] package. BAYESTAR, like other inference pipelines, produces a sky map that contains sky locations for different confidence levels. In Fig. 7.8, we compare the sky location areas for the 90% confidence level for both GstLAL and the

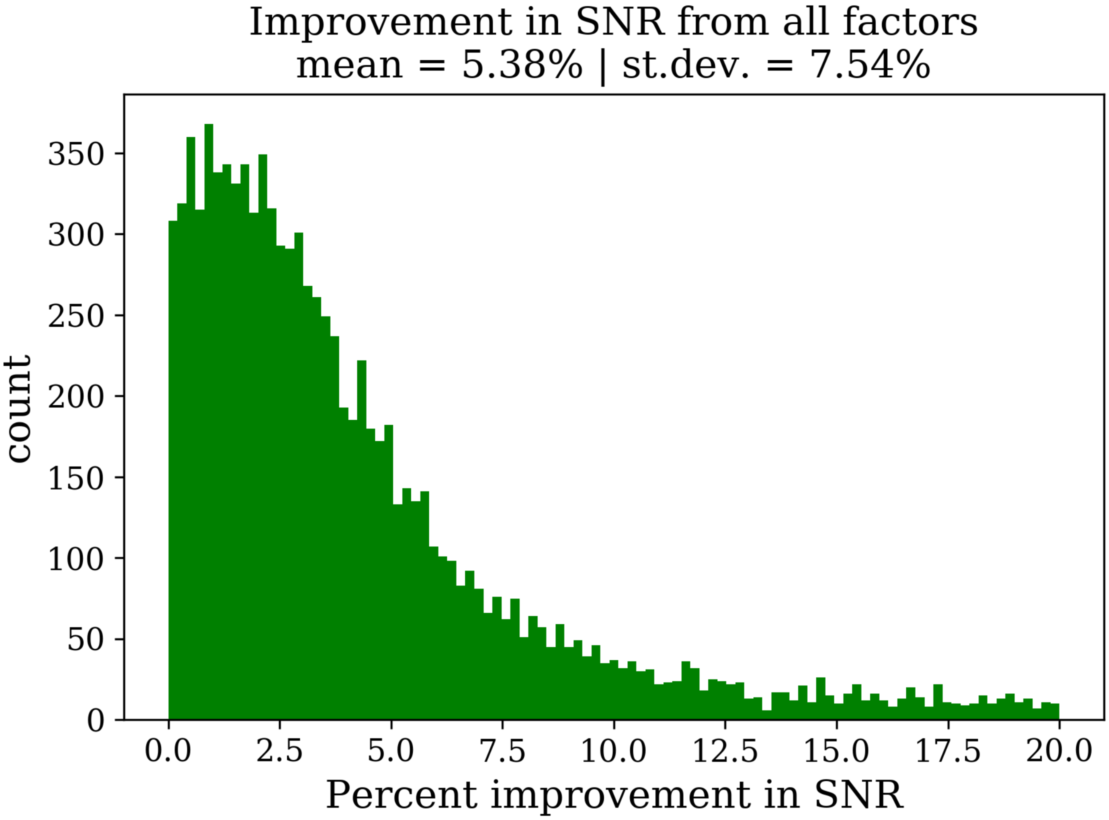


Figure 7.5. A histogram of the combined SNR improvement calculated from the SNR Optimizer uploads as compared to GstLAL uploads.

SNR Optimizer. It shows that the sky maps produced from the SNR Optimizer triggers are on average 16.75% smaller than those produced from GstLAL triggers.

However, a reduction in the sky area by itself only tells us that the SNR Optimizer results are more precise, but not whether they are more accurate. Since these results were calculated from an injection campaign, for which we know the true sky location of all triggers, we can use that to gauge the accuracy of the SNR Optimizer results. To do this, we define two quantities: the searched probability, and the searched area. The searched probability is the minimum confidence level at which the true location of the trigger is still within the sky map. The searched area is the area of the sky map at that confidence level. These can be thought of as a measure of the accuracy of the SNR Optimizer results. The searched area results for GstLAL and the SNR Optimizer are shown in Fig. 7.9. We see that in addition to higher precision, the SNR Optimizer results are also more accurate than those of GstLAL.

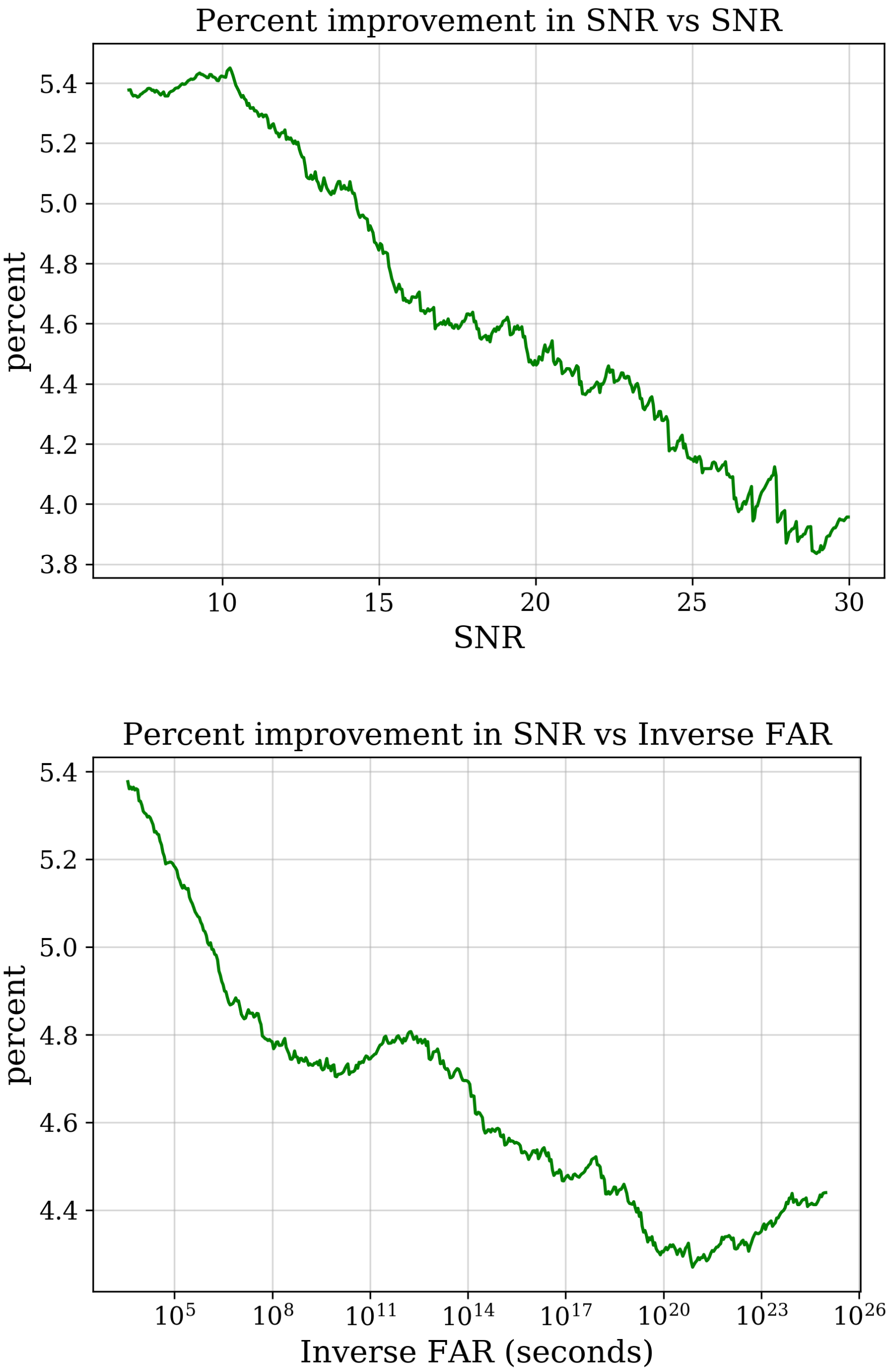


Figure 7.6. This plot shows the complementary cumulative distribution of mean percent improvement in SNR due to the SNR Optimizer vs SNR (top) and inverse FAR (bottom). In both cases, the mean improvement goes down slightly with an increase in SNR or inverse FAR

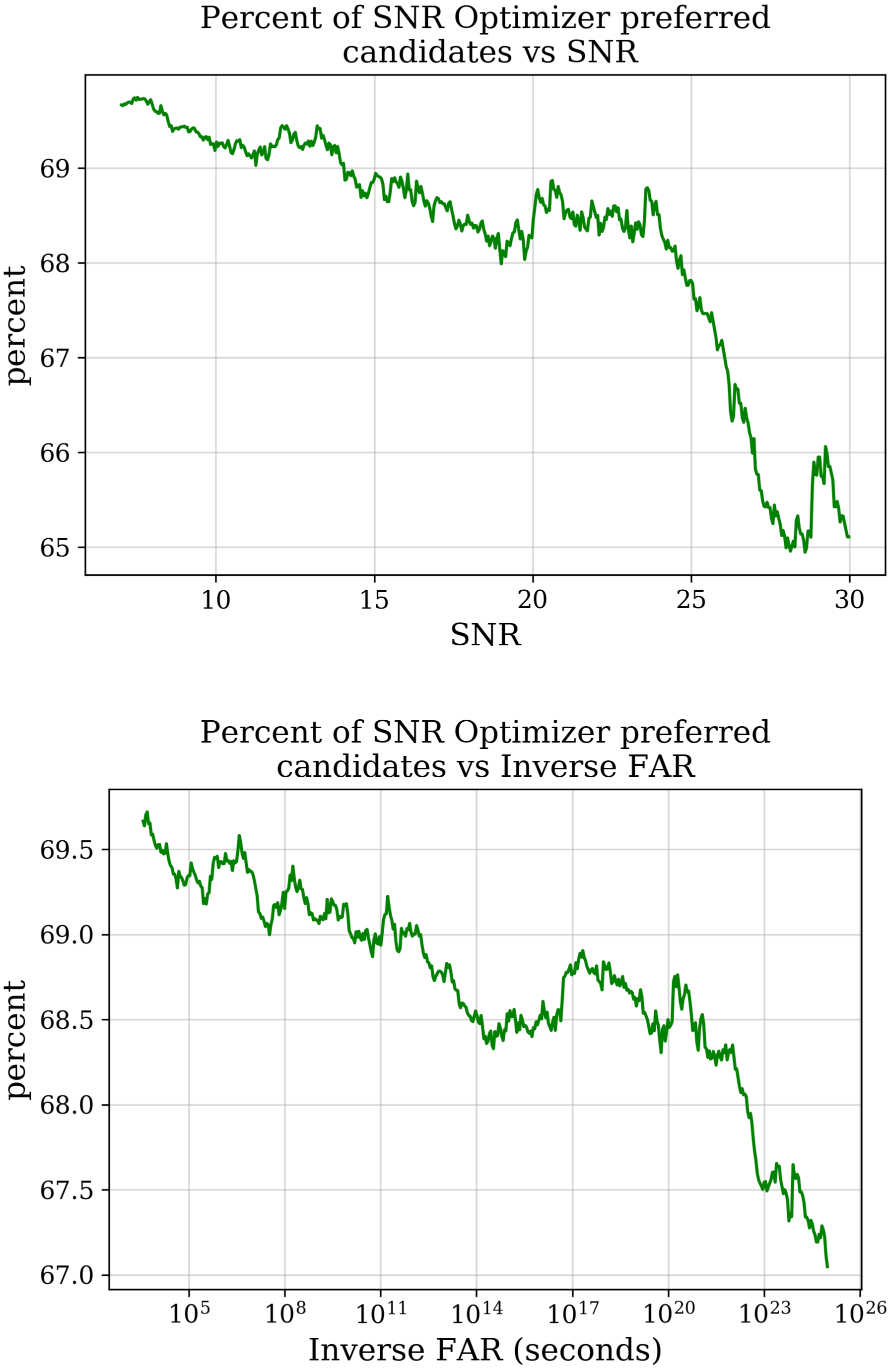


Figure 7.7. This plot shows the complementary cumulative distribution of average percent of times the SNR Optimizer finds a higher SNR than GstLAL (i.e. it is the preferred event) for a candidate vs SNR (top) and inverse FAR (bottom). In both cases, the average preferred event percent goes down slightly with an increase in SNR or inverse FAR

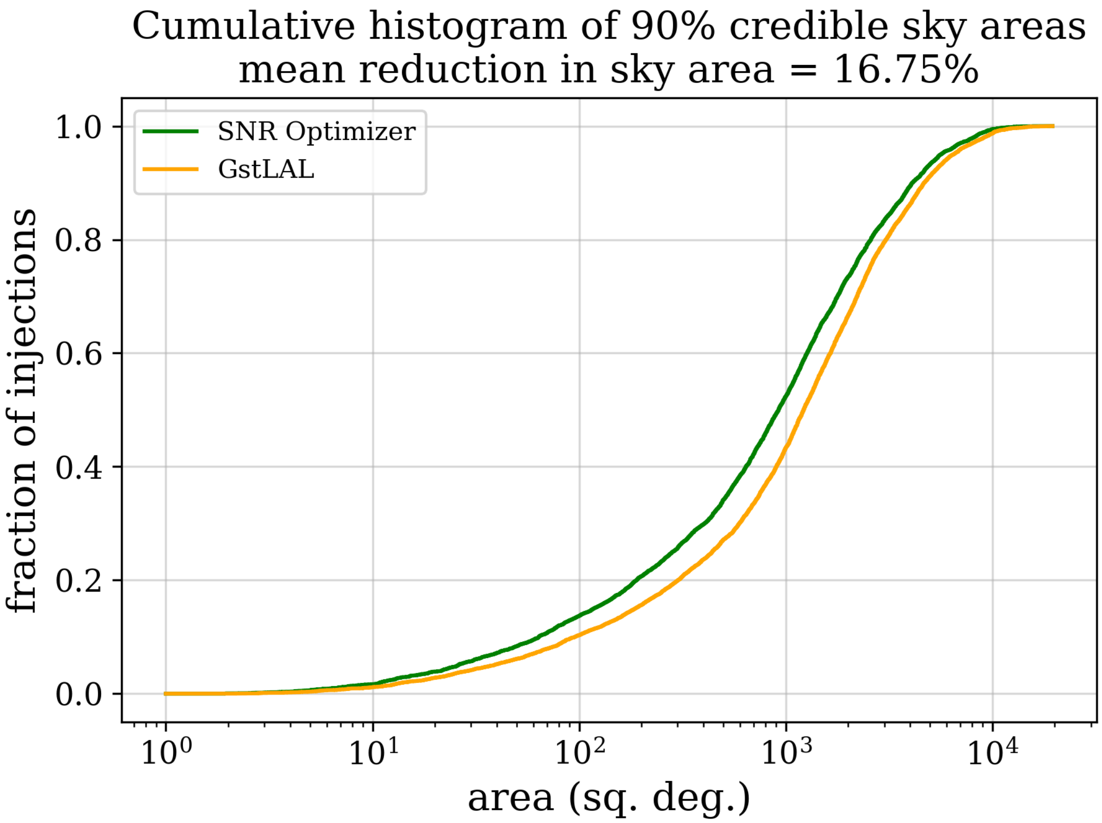


Figure 7.8. This plot shows cumulative histograms of the 90% sky area for GstLAL and the SNR Optimizer. We see that sky maps produced from the SNR Optimizer triggers are on average more constrained than those produced from GstLAL triggers. In other words, the SNR Optimizer results are more precise than those of GstLAL.

Finally, we require a self-consistency condition relating the accuracy and precision results presented above. We require that on average, 90% of triggers have their true location contained within their 90% confidence level sky map, and so on for every confidence level. This test, commonly called a P-P plot, relates the searched probability to the fraction of injections having that value of searched probability. The result is shown in Fig. 7.10, and it shows that the SNR Optimizer results are self-consistent.

7.3.4 Latency

We can calculate two different types of latencies for the SNR Optimizer. The first is the end-to-end latency, which is the time between the GW signal reaching Earth, and the trigger from the SNR Optimizer being uploaded to GraceDB. This includes all sources

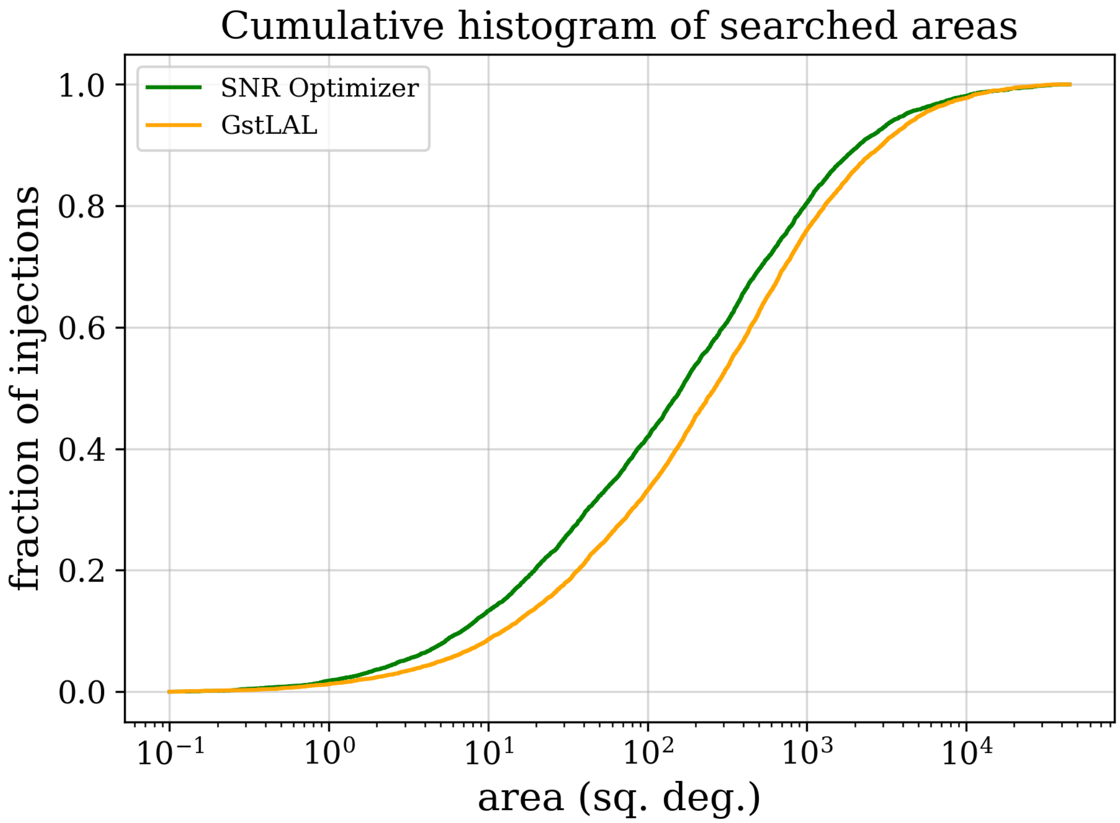


Figure 7.9. This plot shows cumulative histograms of the searched area for GstLAL and the SNR Optimizer. We see that sky maps produced from the SNR Optimizer triggers on average have a lower searched area than those produced from GstLAL triggers. This means we have to go to lower confidence levels for the SNR Optimizer for the sky map to exclude the true location of the trigger, as compared to GstLAL, implying that the SNR Optimizer results are more accurate than those of GstLAL.

of latency, like the data distribution latency, the latency of the initial GstLAL trigger, the internal processing latency of the SNR Optimizer, and any latency incurred by the uploading process. The second type of latency is the internal processing latency recorded by the SNR Optimizer.

A histogram of these two types of latencies, calculated for all SNR Optimizer injection triggers from the MDC is shown in Fig. 7.11. Since the SNR Optimizer has an internal timeout of 240 seconds, we see the histogram for the internal latencies stop at that value. Note that the internal latencies are recorded by the SNR Optimizer jobs, and are not calculated from uploads to GraceDB, and hence the dataset used for this histogram is larger than the set of uploads, because triggers do not get uploaded to GraceDB if they

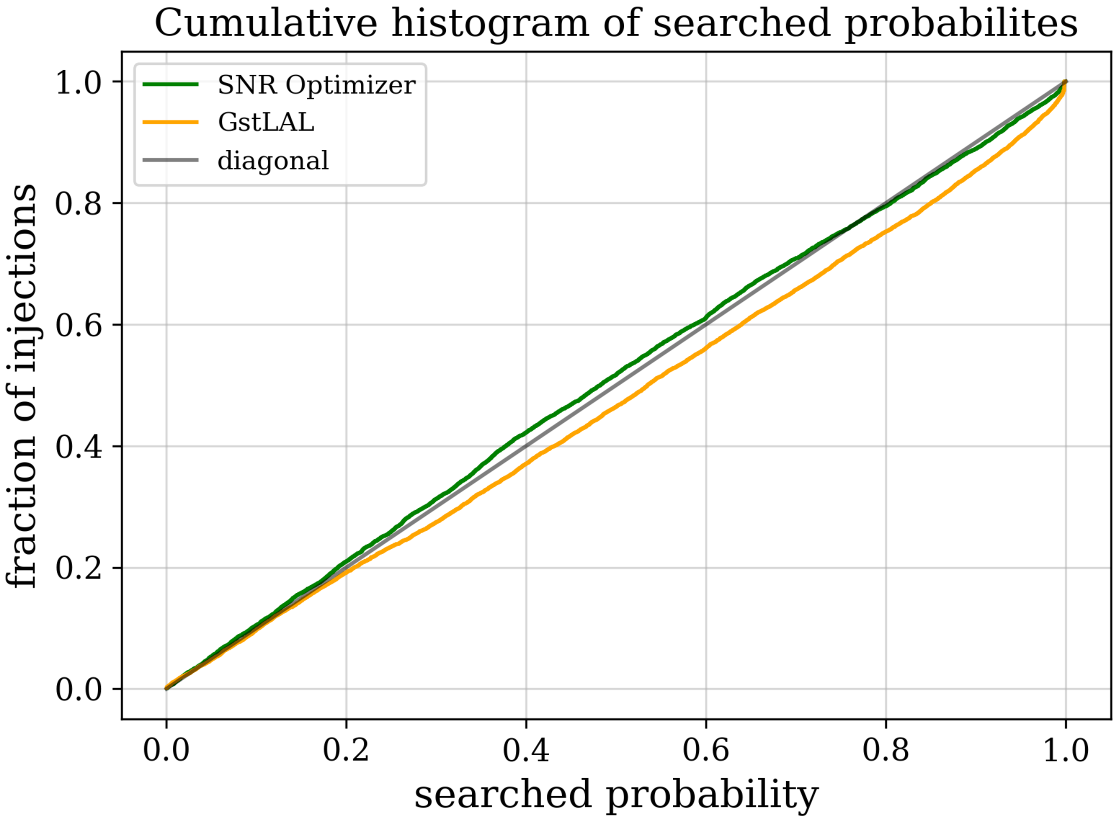
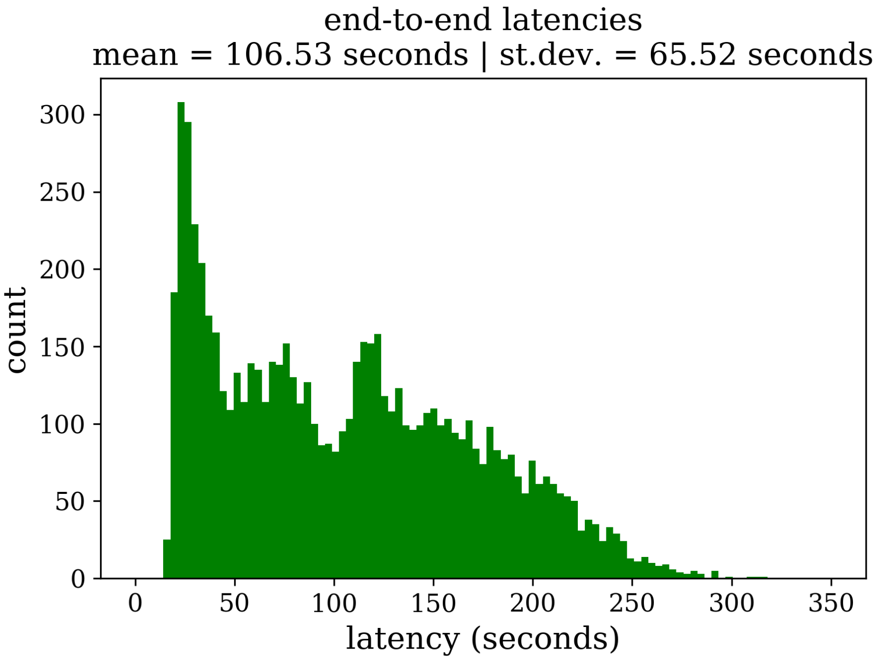


Figure 7.10. This plot shows cumulative histograms of the searched probability for GstLAL and the SNR Optimizer. The SNR Optimizer line is closer to the diagonal than the GstLAL line. This means that the higher precision and accuracy of the SNR Optimzier is also self-consistent. This plot is commonly called a P-P plot.

are not a better trigger than all triggers before. These figures show that the typical latency for the SNR Optimizer is around 100 seconds, and the timeout of 240 seconds is only rarely hit.

7.3.5 Contributions to the SNR improvement

As discussed in Sec. 7.2, the SNR Optimizer finds a higher SNR than GstLAL because of 4 main features: finding a better template, real-time template whitening, higher FFT length, and higher data sampling rate. In order to quantify the effects of each, we set up an offline SNR Optimizer run with all of these features turned off, and then new runs with each feature turned on separately. By comparing the two, we can get an estimate of the contribution of each in the overall SNR improvement. Note that this is a fully offline



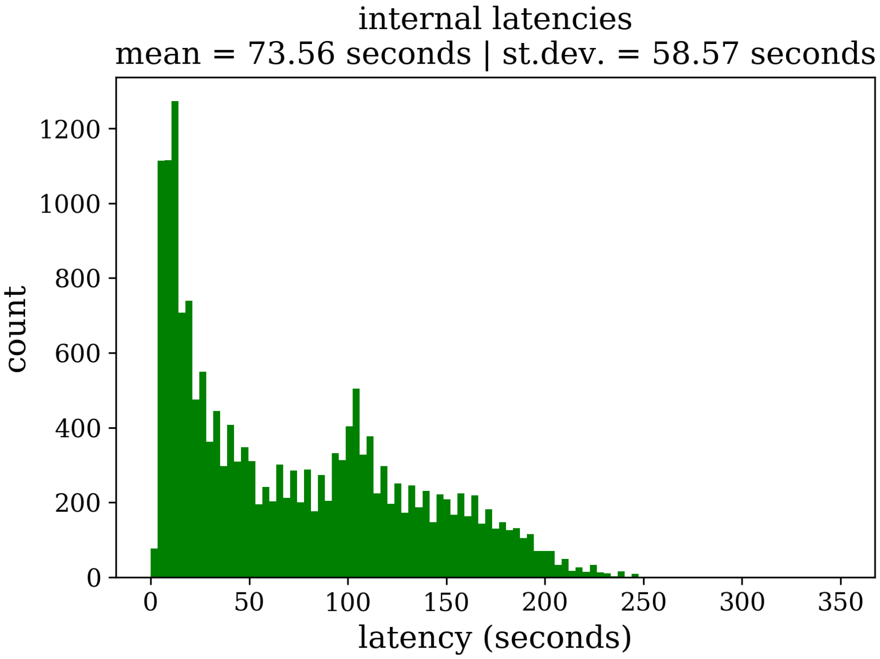


Figure 7.11. The end-to-end latencies (top) and internal latencies (bottom) of the SNR Optimizer. The SNR Optimizer has an internal timeout of 240 seconds, causing the internal latencies to be cut off at that value, and the end-to-end latencies to only rarely exceed that.

test, and hence does not have any effects from the selective uploads performed by an online run. As such, the results here are not directly comparable to the online results presented in the previous subsections.

The result of this test is shown in Fig. 7.12 and Fig. 7.13. We see that finding a better template has on average the highest contribution, at 3.92%, followed by real-time template whitening at 1.41%, higher FFT length at 0.35%, and higher data sampling rate at 0.29%.

7.4 Conclusion

In order to facilitate the multi-messenger follow-up of GW candidates, it is necessary to provide an accurate sky map showing the location of the source of the candidate, so that astronomers know where to point their telescopes. This is done by GW inference pipelines like BAYESTAR or BILBY, which in turn rely on low-latency GW search pipelines like GstLAL, MBTA, PyCBC, and SPIIR, to provide information about the arrival times, phases, and amplitudes of the signal at the different GW detectors.

The estimates of these provided by GW search pipelines might not be completely accurate due to 4 main reasons: discreteness of the template bank used to find signals in the data, lack of template whitening using a real-time PSD, low FFT length used to measure the PSD and whiten the data, and low sampling rate of the data. The latter two are done to reduce latencies of the low-latency analysis. While some of these factors are common to all GW search pipelines, others are specific to GstLAL, the search pipeline that this work focuses on.

We introduced the GstLAL SNR Optimizer as a way to minimize these effects, get higher SNRs and consequently better sky maps in a low-latency GstLAL analysis. It does this by setting up a small targeted, hierarchical, sub-threshold search around GstLAL candidates, in a medium-latency timescale. As long as GstLAL does not report any GW candidates, the SNR Optimizer keeps ingesting data, measuring its PSD, and whitening the data with it. As soon as a candidate is reported, the SNR Optimizer finds the 1000 nearest templates to the template reported by GstLAL, and whitens them with the latest estimate of the PSD it has calculated. It also grabs the the whitened data in a 2 second interval around the time reported by GstLAL, and matched filters this data with the 1000 whitened tempaltes. It retains the 500 highest SNR templates out of these, and splits each of them into two new templates occupying the holes of the previous template bank. To do this, is uses a metric defined on the intrinsic parameter space. As this process

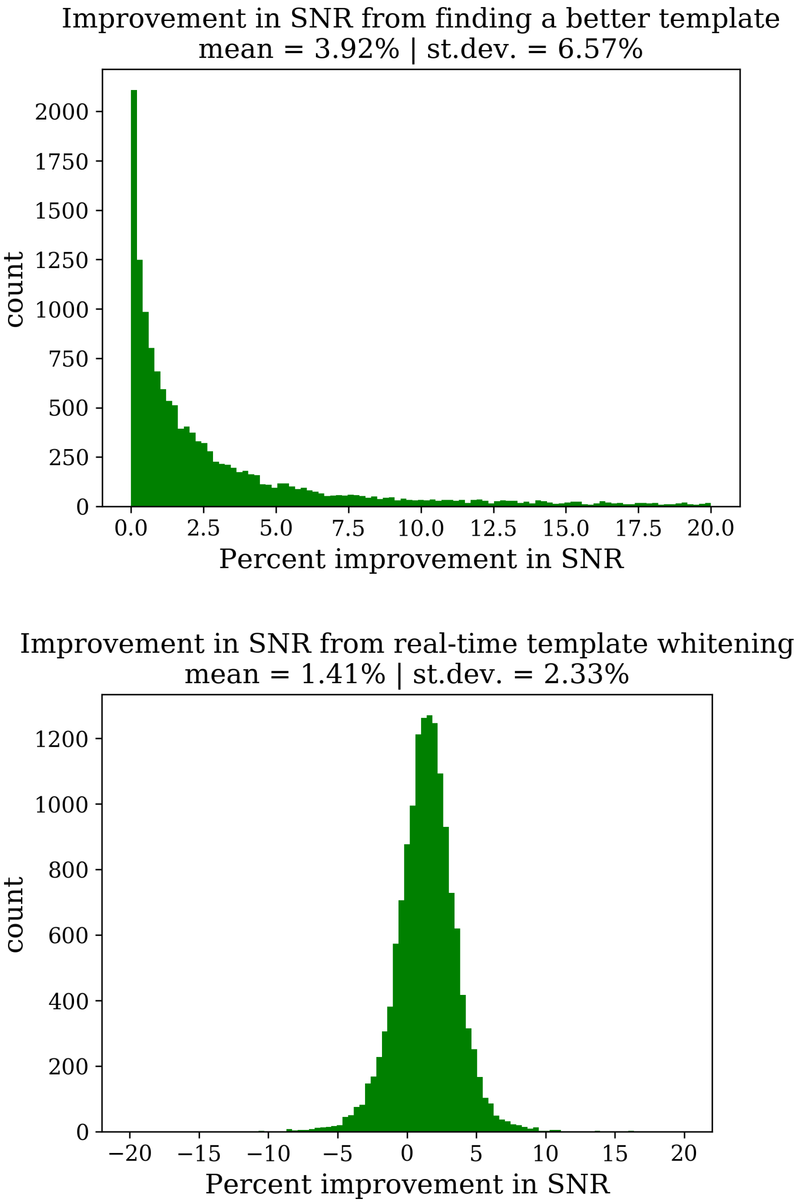


Figure 7.12. This plot shows the contributions from finding a better template (top), and real-time template whitening (bottom), towards the overall SNR gain obtained by the SNR Optimizer. During the process of finding a better template, the templates with the maximum SNR is selected, and hence the contribution from that is strictly positive. Real-time template whitening is only guaranteed to produce positive contributions *on average*. For example, since the PSD can never be perfect due to non-stationary components in the noise, using the PSD measured in real-time to whiten the templates will not always produce a higher SNR, but will do so on average. Fig. 7.13 shows the remaining contributions to SNR gain.

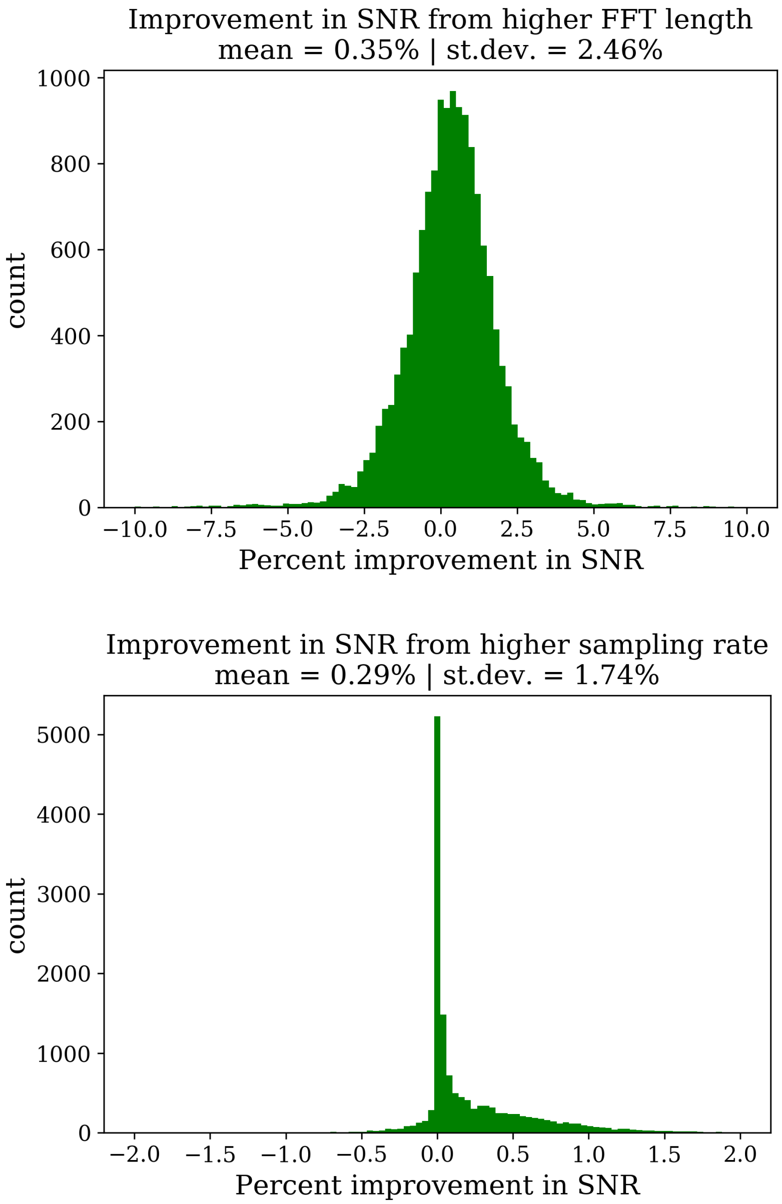


Figure 7.13. This plot is a continuation of Fig. 7.12. This plot shows the contributions from using a higher FFT length (top), and higher data sampling rate (bottom) towards the overall SNR gain obtained by the SNR Optimizer. Similar to what is explained in Fig. 7.12, the higher FFT lenght and data sampling rate are only guaranteed to produce positive contributions *on average*.

keeps repeats, the SNR Optimizer reaches closer and closer to the true parameters of the signal. Templates are not split if their parameter space area is lower than a preset threshold. The process stops when no more templates can split, or an internal timeout of 240 seconds is hit. In this way, the SNR Optimizer is able to find the best matching template for a GW signal. Along with real-time template whiteneing, it can also afford to implement a higher FFT length and a higher data sampling rate, leading to higher SNRs.

To test the efficacy of the SNR Optimizer, we set up a low-latency GstLAL analysis alongside the SNR Optimizer on 40 days of O3 data. We found that the SNR Optimizer is able to find higher SNRs roughly 70% of the time, and that when it does so, it improves the SNR by 5.38% on average. This increase in SNR translates to a 16.75% reduction in the size of the sky map derived from the SNR Optimizer trigger. We also showed that such sky maps are more accurate and self-consistent as compared to those derived from GstLAL triggers. The SNR Optimizer is able to do this with latencies of 100 seconds on average. Finally, we showed that the factor contributing the most to increased SNRs is finding a better template, followed by real-time template whitening, higher FFT length, and higher data sampling rate.

With more electromagnetically bright sources of GWs expected to be detected in the future, tools like the SNR Optimizer are likely to prove useful in facilitating multi-messenger astronomy. GstLAL has already adopted the SNR Optimizer for its low-latency operations during O4.

7.5 Acknowledgements

This research has made use of data, software and/or web tools obtained from the Gravitational Wave Open Science Center (https://www.gw-openscience.org/), a service of LIGO Laboratory, the LIGO Scientific Collaboration (LSC) and the Virgo Collaboration. We especially made heavy use of the LVK Algorithm Library. LIGO was constructed by the California Institute of Technology and the Massachusetts Institute of Technology with funding from the United States National Science Foundation (NSF) and operates under cooperative agreements PHYS-0757058 and PHY-0823459. In addition, the Science and Technology Facilities Council (STFC) of the United Kingdom, the Max-Planck-Society (MPS), and the State of Niedersachsen/Germany supported the construction of Advanced Laser Interferometer Gravitational-Wave Observatory (aLIGO) and construction and operation of the GEO600 detector. Additional support for aLIGO was provided by the Australian Research Council. Virgo is funded, through the European Gravitational Observatory (EGO), by the French Centre National de Recherche Scientifique (CNRS), the Italian Istituto Nazionale di Fisica Nucleare (INFN) and the Dutch Nikhef, with contributions by institutions from Belgium, Germany, Greece, Hungary, Ireland, Japan, Monaco, Poland, Portugal, Spain.

The authors are grateful for computational resources provided by the the LIGO Lab culster at the LIGO Laboratory and supported by PHY-0757058 and PHY−0823459, the Pennsylvania State University's Institute for Computational and Data Sciences gravitational-wave cluster, and supported by OAC-2103662, PHY-2308881, PHY-2011865, OAC-2201445, OAC-2018299, and PHY-2207728. CH Acknowledges generous support from the Eberly College of Science, the Department of Physics, the Institute for Gravitation and the Cosmos, the Institute for Computational and Data Sciences, and the Freed Early Career Professorship.

7.6 Appendix A: Compatibility with other GW search pipelines

While the SNR Optimizer implements a lot of the same ideas as GstLAL, such as matched filtering in the time domain, data and template whitening in the frequency domain, as well as using the `GStreamer` software [169] to stream data, it does so independently of GstLAL. As such, it is designed almost completely modularly and can in theory be used by any other GW search pipeline. Also possible is to have a common set of SNR Optimizer instances listening to uploads from all GW search pipelines.

The initial template bank required by the SNR Optimizer needs to be made by `manifold`, since it needs to contain the parameter space metric information calculated by `manifold`, but this initial template bank does not need to be the same as the template bank used by the search pipeline it is listening to. Currently, there are 2 minor dependencies that the SNR Optimizer has on GstLAL, but both of these are easy to fix:

1. The SNR Optimizer relies on the GstLAL job which uploads triggers to GraceDB to process its own uploads. However, another instance of the same job can be easily set up by the SNR Optimizer to make its uploads independent of the GstLAL uploader job.

2. The SNR Optimizer relies on the `Apache Kafka` software package to get information from the GstLAL low-latency analysis about candidates it found. Since `kafka` is not

necessarily used by all search pipelines, this method of communication can instead by changed to be through GraceDB. There already exists code to continuously communicate with GraceDB about uploads sent to it, and that can be repurposed over here. This method will increase the SNR Optimizer latencies by a small amount, since the communication is not happening locally like in `kafka`, but this increase will only be of the order of a second.

7.7 Appendix B: Low-Latency Mode

Since the SNR Optimizer processes data independently of GstLAL, it can analyze data from more detectors than what GstLAL is doing. This can often happen when a detector like Virgo is deemed to not be sensitive enough to be included in the GstLAL low-latency search, since data from the Virgo detector will be used to calculate the significnace of candidates, potentially leading to a less sensitive analysis. In such situations, the SNR Optimizer can ingest GstLAL's candidates formed from LIGO Hanford and Livingston data, and use that to form candidates with Virgo data added to it as well. This is particularly useful, since adding detectors to a candidate gives by far the biggest improvement in sky maps, as discussed in Sec. 7.1.

In such a use case, since the SNR Optimizer is providing a large amount of new information to the downstream GW inference pipelines, rather than an incremental amount, it is necessary to reduce the SNR Optimizer's latencies to fall in the low-latency regime. As discussed before, the SNR Optimizer normally operates in medium-latency, and has an average latency of 100 seconds. This is because during the hierarchical search process, it matched filters the data with tens of thousands of templates. In order to reduce latencies, a few things can be done:

1. The SNR Optimizer can stop doing any maximization over templates, and to only use the template reported by GstLAL to produce the SNR Optimizer's trigger

2. The SNR Optimizer can re-use the Hanford and Livingston data supplied by GstLAL, and only matched filter Virgo data and add it to the candidate, while still observing the coincidence formation method discussed in Sec. 7.2.

3. The SNR Optimizer can set its FFT length to be low, like GstLAL does (4 seconds worth of data)

4. The SNR Optimizer can set its data sampling rate to be low, like GstLAL does (2048Hz)

By doing so, the SNR Optimizer is able to very quickly add Virgo data to the candidates formed by GstLAL from Hanford and Livingston data. The average internal latency of the SNR Optimzier in this low-latency mode is less than 1 second, with end-to-end latencies averaging around 10 seconds. This low-latency mode of the SNR Optimizer was used alongside the regular mode by GstLAL for part of its O4 low-latency operations.

Bibliography

[1] Sathyaprakash, B. S. and B. F. Schutz (2009) "Physics, Astrophysics and Cosmology with Gravitational Waves," *Living Reviews in Relativity*, **12**(1).
URL http://dx.doi.org/10.12942/lrr-2009-2

[2] LIGO-India, "LIGO-India: Detector," .
URL https://www.ligo-india.in/activities/research/detector/

[3] Gravitational Wave Open Science Center, "O3a Data Release," .
URL https://gwosc.org/O3/O3a/

[4] Joshi, P. et al. (2024), "How Many Times Should We Matched Filter Gravitational Wave Data? A Comparison of GstLAL's Online and Offline Performance," In preparation.

[5] Chaudhary, S. S. et al. (2023), "Low-latency alert products and their performance in anticipation of the fourth LIGO-Virgo-KAGRA observing run," (in prep).

[6] Abbott, R. et al. (2021), "GWTC-3: Compact Binary Coalescences Observed by LIGO and Virgo During the Second Part of the Third Observing Run," 2111.03606.

[7] (2021) "GWTC-2: Compact Binary Coalescences Observed by LIGO and Virgo during the First Half of the Third Observing Run," *Physical Review X*, **11**(2), p. 021053.

[8] The LIGO Scientific Collaboration, The Virgo Collaboration, R. Abbott, et al. (2021), "GWTC-2.1: Deep Extended Catalog of Compact Binary Coalescences Observed by LIGO and Virgo During the First Half of the Third Observing Run," .
URL https://arxiv.org/abs/2108.01045

[9] Flanagan, E. E. and S. A. Hughes (2005) "The basics of gravitational wave theory," *New Journal of Physics*, **7**, p. 204–204.
URL http://dx.doi.org/10.1088/1367-2630/7/1/204

[10] Creighton, J. D. and W. G. Anderson (2012) *Gravitational-wave physics and astronomy: An introduction to theory, experiment and data analysis*, John Wiley & Sons.

[11] Aasi, J. et al. (2015) "Advanced LIGO," *Class. Quant. Grav.*, **32**, p. 074001, `1411.4547`.

[12] Acernese, F. et al. (2015) "Advanced Virgo: a second-generation interferometric gravitational wave detector," *Class. Quant. Grav.*, **32**(2), p. 024001, `1408.3978`.

[13] Akutsu, T. et al. (2020) "Overview of KAGRA: Detector design and construction history," *Progress of Theoretical and Experimental Physics*, **2021**(5), 05A101, `https://academic.oup.com/ptep/article-pdf/2021/5/05A101/37974994/ptaa125.pdf`.
URL `https://doi.org/10.1093/ptep/ptaa125`

[14] Abbott, B. P. et al. (2019) "GWTC-1: A Gravitational-Wave Transient Catalog of Compact Binary Mergers Observed by LIGO and Virgo during the First and Second Observing Runs," *Phys. Rev. X*, **9**(3), p. 031040, `1811.12907`.

[15] 245, D. C. H. J. . K. V. . R. D. . T. L. . . S. D. . V. S. . Y. S. . . and L. C. O. C. A. I. . . H. G. . . H. D. A. . . M. C. . . P. D. . V. S. . 247 (2017) "A gravitational-wave standard siren measurement of the Hubble constant," *Nature*, **551**(7678), pp. 85–88.

[16] Abbott, R., H. Abe, F. Acernese, K. Ackley, N. Adhikari, R. Adhikari, V. Adkins, V. Adya, C. Affeldt, D. Agarwal, et al. (2021) "Tests of general relativity with GWTC-3," *arXiv preprint arXiv:2112.06861*.

[17] Abbott, R., T. Abbott, S. Abraham, F. Acernese, K. Ackley, C. Adams, R. Adhikari, V. Adya, C. Affeldt, M. Agathos, et al. (2020) "Properties and astrophysical implications of the $150M_\odot$ binary black hole merger GW190521," *The Astrophysical Journal Letters*, **900**(1), p. L13.

[18] Owen, B. J. and B. S. Sathyaprakash (1999) "Matched filtering of gravitational waves from inspiraling compact binaries: Computational cost and template placement," *Phys. Rev. D*, **60**, p. 022002.
URL `https://link.aps.org/doi/10.1103/PhysRevD.60.022002`

[19] Messick, C. et al. (2017) "Analysis Framework for the Prompt Discovery of Compact Binary Mergers in Gravitational-wave Data," *Phys. Rev. D*, **95**(4), p. 042001, `1604.04324`.

[20] Cannon, K. et al. (2020), "GstLAL: A software framework for gravitational wave discovery," `2010.05082`.

[21] Sachdev, S. et al. (2019), "The GstLAL Search Analysis Methods for Compact Binary Mergers in Advanced LIGO's Second and Advanced Virgo's First Observing Runs," `1901.08580`.

[22] Hanna, C. et al. (2020) "Fast evaluation of multi-detector consistency for real-time gravitational wave searches," *Phys. Rev.*, **D101**(2), p. 022003, `1901.02227`.

[23] Tsukada, L., P. Joshi, S. Adhicary, R. George, A. Guimaraes, C. Hanna, R. Magee, A. Zimmerman, P. Baral, A. Baylor, K. Cannon, S. Caudill, B. Cousins, J. D. E. Creighton, B. Ewing, H. Fong, P. Godwin, R. Harada, Y.-J. Huang, R. Huxford, J. Kennington, S. Kuwahara, A. K. Y. Li, D. Meacher, C. Messick, S. Morisaki, D. Mukherjee, W. Niu, A. Pace, C. Posnansky, A. Ray, S. Sachdev, S. Sakon, D. Singh, R. Tapia, T. Tsutsui, K. Ueno, A. Viets, L. Wade, and M. Wade (2023) "Improved ranking statistics of the GstLAL inspiral search for compact binary coalescences," `2305.06286`.

[24] Cannon, K., C. Hanna, and J. Peoples (2015), "Likelihood-Ratio Ranking Statistic for Compact Binary Coalescence Candidates with Rate Estimation," `1504.04632`.

[25] Cannon, K., C. Hanna, and D. Keppel (2013) "Method to estimate the significance of coincident gravitational-wave observations from compact binary coalescence," *Phys. Rev. D*, **88**(2), p. 024025, `1209.0718`.

[26] Canton, T. D., A. H. Nitz, B. Gadre, G. S. C. Davies, V. Villa-Ortega, T. Dent, I. Harry, and L. Xiao (2021) "Real-time Search for Compact Binary Mergers in Advanced LIGO and Virgo's Third Observing Run Using PyCBC Live," *The Astrophysical Journal*, **923**(2), p. 254.
URL `https://dx.doi.org/10.3847/1538-4357/ac2f9a`

[27] Davies, G. S., T. Dent, M. Tápai, I. Harry, C. McIsaac, and A. H. Nitz (2020) "Extending the PyCBC search for gravitational waves from compact binary mergers to a global network," *Physical Review D*, **102**(2).
URL `https://doi.org/10.1103%2Fphysrevd.102.022004`

[28] Nitz, A. H. (2018) "Rapid detection of gravitational waves from compact binary mergers with PyCBC Live," *Physical Review D*, **98**(2).

[29] Aubin, F., F. Brighenti, R. Chierici, D. Estevez, G. Greco, G. M. Guidi, V. Juste, F. Marion, B. Mours, E. Nitoglia, O. Sauter, and V. Sordini (2021) "The MBTA pipeline for detecting compact binary coalescences in the third LIGO–Virgo observing run," *Classical and Quantum Gravity*, **38**(9), p. 095004.
URL `https://doi.org/10.1088%2F1361-6382%2Fabe913`

[30] Adams, T., D. Buskulic, V. Germain, G. M. Guidi, F. Marion, M. Montani, B. Mours, F. Piergiovanni, and G. Wang (2016) "Low-latency analysis pipeline for compact binary coalescences in the advanced gravitational wave detector era," *Classical and Quantum Gravity*, **33**(17), p. 175012.
URL `https://dx.doi.org/10.1088/0264-9381/33/17/175012`

[31] Chu, Q. (2022) "SPIIR online coherent pipeline to search for gravitational waves from compact binary coalescences," *Physical Review D*, **105**(2).

[32] ——— (2017) *Low-latency detection and localization of gravitational waves from compact binary coalescences*, Ph.D. thesis, The University of Western Australia.

[33] Venumadhav, T., B. Zackay, J. Roulet, L. Dai, and M. Zaldarriaga (2019) "New search pipeline for compact binary mergers: Results for binary black holes in the first observing run of Advanced LIGO," *Phys. Rev. D*, **100**, p. 023011.
URL https://link.aps.org/doi/10.1103/PhysRevD.100.023011

[34] Zackay, B., L. Dai, T. Venumadhav, J. Roulet, and M. Zaldarriaga (2021) "Detecting gravitational waves with disparate detector responses: Two new binary black hole mergers," *Physical Review D*, **104**(6).
URL https://doi.org/10.1103%2Fphysrevd.104.063030

[35] Moe, B., P. Brady, B. Stephens, E. Katsavounidis, R. Williams, and F. Zhang (2014), "GraceDB: A Gravitational Wave Candidate Event Database," .
URL https://dcc.ligo.org/LIGO-T1400365

[36] LIGO Scientific Collaboration and Virgo Collaboration (2017) *GCN*, **21505**.
URL https://gcn.gsfc.nasa.gov/other/G298048.gcn3

[37] Abbott, B. P. et al. (2017) "GW170817: Observation of Gravitational Waves from a Binary Neutron Star Inspiral," *Phys. Rev. Lett.*, **119**(16), p. 161101, 1710.05832.

[38] ——— (2018) "GW170817: Measurements of Neutron Star Radii and Equation of State," *Phys. Rev. Lett.*, **121**, p. 161101.
URL https://link.aps.org/doi/10.1103/PhysRevLett.121.161101

[39] ——— (2019) "Tests of General Relativity with GW170817," *Phys. Rev. Lett.*, **123**, p. 011102.
URL https://link.aps.org/doi/10.1103/PhysRevLett.123.011102

[40] Hanna, C., J. Kennington, S. Sakon, S. Privitera, M. Fernandez, J. Wang, C. Messick, A. Pace, K. Cannon, P. Joshi, R. Huxford, S. Caudill, C. Chan, B. Cousins, J. D. E. Creighton, B. Ewing, H. Fong, P. Godwin, R. Magee, D. Meacher, S. Morisaki, D. Mukherjee, H. Ohta, S. Sachdev, D. Singh, R. Tapia, L. Tsukada, D. Tsuna, T. Tsutsui, K. Ueno, A. Viets, L. Wade, and M. Wade (2023) "Binary tree approach to template placement for searches for gravitational waves from compact binary mergers," *Phys. Rev. D*, **108**, p. 042003.
URL https://link.aps.org/doi/10.1103/PhysRevD.108.042003

[41] Sakon, S. et al. (2022), "Template bank for compact binary mergers in the fourth observing run of Advanced LIGO, Advanced Virgo, and KAGRA," 2211.16674.

[42] Joshi, P., L. Tsukada, and C. Hanna (2023), "Background Filter: A method for removing signal contamination during significance estimation of a GstLAL anaysis," 2305.18233.

[43] Ewing, B., R. Huxford, D. Singh, L. Tsukada, C. Hanna, Y.-J. Huang, P. Joshi, A. K. Y. Li, R. Magee, C. Messick, A. Pace, A. Ray, S. Sachdev, S. Sakon, R. Tapia, S. Adhicary, P. Baral, A. Baylor, K. Cannon, S. Caudill, S. S. Chaudhary, M. W. Coughlin, B. Cousins, J. D. E. Creighton, R. Essick, H. Fong, R. N. George, P. Godwin, R. Harada, J. Kennington, S. Kuwahara, D. Meacher, S. Morisaki, D. Mukherjee, W. Niu, C. Posnansky, A. Toivonen, T. Tsutsui, K. Ueno, A. Viets, L. Wade, M. Wade, and G. Waratkar (2023) "Performance of the low-latency GstLAL inspiral search towards LIGO, Virgo, and KAGRA's fourth observing run," 2305.05625.

[44] "IGWN Public Alerts User Guide," https://emfollow.docs.ligo.org/userguide/analysis/index.html##alert-threshold.

[45] Singer, L. P. and L. R. Price (2016), "Rapid Bayesian position reconstruction for gravitational-wave transients," 1508.03634.

[46] Singer, L. P. et al. (2016) "Going the Distance: Mapping Host Galaxies of LIGO and Virgo Sources in Three Dimensions Using Local Cosmography and Targeted Follow-up," *Astrophys. J. Lett.*, **829**(1), p. L15, 1603.07333.

[47] Ashton, G. et al. (2019) "BILBY: A user-friendly Bayesian inference library for gravitational-wave astronomy," *Astrophys. J. Suppl.*, **241**(2), p. 27, 1811.02042.

[48] Romero-Shaw, I. M. et al. (2020) "Bayesian inference for compact binary coalescences with bilby: validation and application to the first LIGO–Virgo gravitational-wave transient catalogue," *Mon. Not. Roy. Astron. Soc.*, **499**(3), pp. 3295–3319, 2006.00714.

[49] NASA *GCN*.
URL https://gcn.gsfc.nasa.gov/about.html

[50] Rose, C. A. (2024) *Rapid Parameter Estimation of Compact Binary Coalescences with Gravitational Waves*, Ph.D. thesis, The University of Wisconsin-Milwaukee.

[51] Ray, A. et al. (2023), "When to Point Your Telescopes: Gravitational Wave Trigger Classification for Real-Time Multi-Messenger Followup Observations," 2306.07190.

[52] Villa-Ortega, V., T. Dent, and A. C. Barroso (2022) "Astrophysical Source Classification and Distance Estimation for PyCBC Live," *arXiv preprint arXiv:2203.10080*.

[53] Andres, N., M. Assiduo, F. Aubin, R. Chierici, D. Estevez, F. Faedi, G. M. Guidi, V. Juste, F. Marion, B. Mours, E. Nitoglia, and V. Sordini (2022) "Assessing the compact-binary merger candidates reported by the MBTA pipeline in the LIGO–Virgo O3 run: probability of astrophysical origin, classification, and associated uncertainties," *Classical and Quantum Gravity*, **39**(5), p. 055002.
URL `https://dx.doi.org/10.1088/1361-6382/ac482a`

[54] Piotrzkowski, B. (2022) *Searching for Gravitational Wave Associations with High-Energy Astrophysical Transients*, Ph.D. thesis, The University of Wisconsin-Milwaukee.

[55] Urban, A. L. (2016) *Monsters in the dark: High energy signatures of black hole formation with multimessenger astronomy*, Ph.D. thesis, The University of Wisconsin-Milwaukee.

[56] Cho, M.-A. (2019) *Low-latency searches for gravitational waves and their electromagnetic counterparts with Advanced LIGO and Virgo*, Ph.D. thesis, University of Maryland, College Park.

[57] Collaboration, T. L. S. and the Virgo Collaboration (2021) "Population Properties of Compact Objects from the Second LIGO–Virgo Gravitational-Wave Transient Catalog," *The Astrophysical Journal Letters*, **913**(1), p. L7.
URL `https://dx.doi.org/10.3847/2041-8213/abe949`

[58] Abbott, R. et al. (2023) "Population of Merging Compact Binaries Inferred Using Gravitational Waves through GWTC-3," *Phys. Rev. X*, **13**(1), p. 011048, `2111.03634`.

[59] Abbott, B. P. et al. (2019) "Tests of general relativity with the binary black hole signals from the LIGO-Virgo catalog GWTC-1," *Phys. Rev. D*, **100**, p. 104036.
URL `https://link.aps.org/doi/10.1103/PhysRevD.100.104036`

[60] Abbott, R. et al. (2021) "Tests of general relativity with binary black holes from the second LIGO-Virgo gravitational-wave transient catalog," *Phys. Rev. D*, **103**, p. 122002.
URL `https://link.aps.org/doi/10.1103/PhysRevD.103.122002`

[61] Abbott, B. P. et al. (2016) "Tests of General Relativity with GW150914," *Phys. Rev. Lett.*, **116**, p. 221101.
URL `https://link.aps.org/doi/10.1103/PhysRevLett.116.221101`

[62] Magee, R., A.-S. Deutsch, P. McClincy, C. Hanna, C. Horst, D. Meacher, C. Messick, S. Shandera, and M. Wade (2018) "Methods for the detection of gravitational waves from subsolar mass ultracompact binaries," *Phys. Rev. D*, **98**, p. 103024.
URL `https://link.aps.org/doi/10.1103/PhysRevD.98.103024`

[63] Cannon, K. et al. (2012) "Toward Early-Warning Detection of Gravitational Waves from Compact Binary Coalescence," *Astrophys. J.*, **748**, p. 136, 1107.2665.

[64] Sachdev, S., R. Magee, C. Hanna, K. Cannon, L. Singer, J. R. Sk, D. Mukherjee, S. Caudill, C. Chan, J. D. Creighton, et al. (2020) "An early-warning system for electromagnetic follow-up of gravitational-wave events," *The Astrophysical Journal Letters*, **905**(2), p. L25.

[65] Magee, R., D. Chatterjee, L. P. Singer, S. Sachdev, M. Kovalam, G. Mo, S. Anderson, P. Brady, P. Brockill, K. Cannon, et al. (2021) "First demonstration of early warning gravitational-wave alerts," *The Astrophysical Journal Letters*, **910**(2), p. L21.

[66] Singh, D. (2024) *Exploring Dark Matter Through Gravitational-Wave Observations*, Ph.D. thesis, The Pennsylvania State University.

[67] Hanna, C., J. Kennington, W. Niu, S. Sakon, D. Singh, S. Adhicary, P. Baral, A. Baylor, K. Cannon, S. Caudill, B. Cousins, J. D. E. Creighton, B. Ewing, H. Fong, R. N. George, P. Godwin, R. Harada, Y.-J. Huang, R. Huxford, P. Joshi, S. Kuwahara, A. K. Y. Li, R. Magee, D. Meacher, C. Messick, S. Morisaki, D. Mukherjee, A. Pace, C. Posnansky, A. Ray, S. Sachdev, S. Schmidt, U. Shah, R. Tapia, L. Tsukada, K. Ueno, A. Viets, L. Wade, M. Wade, Z. Yarbrough, and N. Zhang (2024), "Template bank for sub solar mass compact binary mergers in the fourth observing run of Advanced LIGO, Advanced Virgo, and KAGRA," 2412.10951.
URL https://arxiv.org/abs/2412.10951

[68] Sachdev, S., R. Magee, C. Hanna, R. Ewing, D. Singh, R. Huxford, L. Tsukada, S. Sakon, P. Joshi, A. Ray, W. Niu, A. Baylor, P. Baral, and A. Li (2023), "GstLAL EW Online Documents," LIGO Document T2200352-v7.
URL https://dcc.ligo.org/LIGO-T2200352/public

[69] LIGO, Virgo, and KAGRA Collaborations (2025), "LIGO, Virgo, and KAGRA Observing Run Plans," Accessed: 2025-02-06.
URL https://observing.docs.ligo.org/plan/

[70] Klimenko, S., G. Vedovato, M. Drago, F. Salemi, V. Tiwari, G. Prodi, C. Lazzaro, K. Ackley, S. Tiwari, C. Da Silva, et al. (2016) "Method for detection and reconstruction of gravitational wave transients with networks of advanced detectors," *Physical Review D*, **93**(4), p. 042004.

[71] Klimenko, S., S. Mohanty, M. Rakhmanov, and G. Mitselmakher (2005) "Constraint likelihood analysis for a network of gravitational wave detectors," *Physical Review D—Particles, Fields, Gravitation, and Cosmology*, **72**(12), p. 122002.

[72] Lynch, R., S. Vitale, R. Essick, E. Katsavounidis, and F. Robinet (2017) "Information-theoretic approach to the gravitational-wave burst detection problem," *Physical Review D*, **95**(10), p. 104046.

[73] Joshi, P. et al. (2024), "Method to get Better Sky Maps in a GstLAL Low-Latency Analysis," In preparation.

[74] Huxford, R. (2024) *Towards Exclusive Low-Latency Detection of Gravitational Waves and the Inference of Neutron Star Equation of State With Next Generation Detection Networks*, Ph.D. thesis, The Pennsylvania State University.

[75] Ewing, B. (2024) *Gravitational Wave Searches Towards Multi-Messenger and Multi-Band Astronomy*, Ph.D. thesis, The Pennsylvania State University.

[76] Collaboration, L. S., V. Collaboration, K. Collaboration, et al. (2024) "Observation of Gravitational Waves from the Coalescence of a 2.5–4.5 $M_\odot$ Compact Object and a Neutron Star," *The Astrophysical Journal Letters*, **970**(2), p. L34.
URL `http://dx.doi.org/10.3847/2041-8213/ad5beb`

[77] LVC (2023), "GRB 230529A: LIGO/Virgo detection of gravitational waves associated with a gamma-ray burst," `https://gcn.gsfc.nasa.gov/notices_l/S230529ay.lvc`.

[78] Collaboration, T. L. S., T. V. Collaboration, and T. K. Collaboration (2023), "GCN CIRCULAR NUMBER: 33891, LIGO/Virgo/KAGRA S230529ay: Updated Sky localization and EM Bright Classification," `https://gcn.nasa.gov/circulars/33891.txt`, accessed: 2025-02-06.

[79] Ligo Scientific Collaboration, VIRGO Collaboration, and Kagra Collaboration (2025) "LIGO/Virgo/KAGRA S250109bi: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38863**, p. 1.

[80] ——— (2024) "LIGO/Virgo/KAGRA S241230ev: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38735**, p. 1.

[81] ——— (2024) "LIGO/Virgo/KAGRA S241230bd: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38722**, p. 1.

[82] ——— (2024) "LIGO/Virgo/KAGRA S241225c: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38666**, p. 1.

[83] ——— (2024) "LIGO/Virgo/KAGRA S241210d: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38512**, p. 1.

[84] ——— (2024) "LIGO/Virgo/KAGRA S241201ac: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38402**, p. 1.

[85] ——— (2024) "LIGO/Virgo/KAGRA S241122a: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38301**, p. 1.

[86] ——— (2024) "LIGO/Virgo/KAGRA S241114bi: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38228**, p. 1.

[87] ——— (2024) "LIGO/Virgo/KAGRA S241113p: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38204**, p. 1.

[88] ——— (2024) "LIGO/Virgo/KAGRA S241110br: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38155**, p. 1.

[89] ——— (2024) "LIGO/Virgo/KAGRA S241109p: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **38138**, p. 1.

[90] ——— (2024) "LIGO/Virgo/KAGRA S241101ee: Identification of a GW compact binary merger candidate," GCN Circular 38036.
URL `https://gcn.nasa.gov/circulars/38036?view=index\&query=S241101ee\&startDate=\&endDate=\&sort=circularID`

[91] ——— (2024) "LIGO/Virgo/KAGRA S241011k: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37776**, p. 1.

[92] ——— (2024) "LIGO/Virgo/KAGRA S241009em: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37750**, p. 1.

[93] ——— (2024) "LIGO/Virgo/KAGRA S241009l: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37743**, p. 1.

[94] ——— (2024) "LIGO/Virgo/KAGRA S241002e: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37663**, p. 1.

[95] ——— (2024) "LIGO/Virgo/KAGRA S240921cw: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37586**, p. 1.

[96] ——— (2024) "LIGO/Virgo/KAGRA S240917cb: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37534**, p. 1.

[97] ——— (2024) "LIGO/Virgo/KAGRA S240916ar: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37530**, p. 1.

[98] ——— (2024) "LIGO/Virgo/KAGRA S240915bd: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37514**, p. 1.

[99] ——— (2024) "LIGO/Virgo/KAGRA S240813d: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37170**, p. 1.

[100] ——— (2024) "LIGO/Virgo/KAGRA S240813c: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37168**, p. 1.

[101] ——— (2024) "S240807h: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **37096**, p. 1.

[102] ——— (2024) "LIGO/Virgo/KAGRA S240716b: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36879**, p. 1.

[103] ——— (2024) "LIGO/Virgo/KAGRA S240703ad: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36816**, p. 1.

[104] ——— (2024) "LIGO/Virgo/KAGRA S240627by: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36772**, p. 1.

[105] ——— (2024) "LIGO/Virgo/KAGRA S240525p: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36539**, p. 1.

[106] ——— (2024) "LIGO/Virgo/KAGRA S240514c: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36473**, p. 1.

[107] ——— (2024) "LIGO/Virgo/KAGRA S240428dr: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36340**, p. 1.

[108] ——— (2024) "LIGO/Virgo/KAGRA S240422ed: Identification of a GW compact binary merger candidate," *GRB Coordinates Network*, **36236**, p. 1.

[109] Abbott, B. P. et al. (2016) "Observation of Gravitational Waves from a Binary Black Hole Merger," *Phys. Rev. Lett.*, **116**(6), p. 061102, 1602.03837.

[110] Collaboration, L. (2024), "LIGO-Virgo-KAGRA Cumulative Detection plot - O1-O4b," .
URL https://dcc.ligo.org/LIGO-G2302098-v21/public

[111] Abbott, R. et al. (2021) "Observation of Gravitational Waves from Two Neutron Star–Black Hole Coalescences," *Astrophys. J. Lett.*, **915**(1), p. L5, 2106.15163.

[112] Yang, S.-C., W.-B. Han, and G. Wang (2020) "Tests of weak equivalence principle with the gravitational wave signals in the LIGO–Virgo catalogue GWTC-1," *Monthly Notices of the Royal Astronomical Society: Letters*, **499**(1), pp. L53–L57, https://academic.oup.com/mnrasl/article-pdf/499/1/L53/54638219/mnrasl_499_1_l53.pdf.
URL https://doi.org/10.1093/mnrasl/slaa143

[113] Ezquiaga, J. M. (2021) "Hearing gravity from the cosmos: GWTC-2 probes general relativity at cosmological scales," *Physics Letters B*, **822**, p. 136665.
URL https://www.sciencedirect.com/science/article/pii/S0370269321006055

[114] Collaboration, T. L. S. and the Virgo Collaboration (2021) “A Gravitational-wave Measurement of the Hubble Constant Following the Second Observing Run of Advanced LIGO and Virgo,” *The Astrophysical Journal*, **909**(2), p. 218.
URL `https://dx.doi.org/10.3847/1538-4357/abdcb7`

[115] Wu, S., Z. Cao, and Z.-H. Zhu (2020) “Measuring the eccentricity of binary black holes in GWTC-1 by using the inspiral-only waveform,” *Monthly Notices of the Royal Astronomical Society*, **495**(1), pp. 466–478, `https://academic.oup.com/mnras/article-pdf/495/1/466/33387647/staa1176.pdf`.
URL `https://doi.org/10.1093/mnras/staa1176`

[116] Wang, H.-T., P.-C. Li, J.-L. Jiang, G.-W. Yuan, Y.-M. Hu, and Y.-Z. Fan (2021) “Constrains on the electric charges of the binary black holes with GWTC-1 events,” *The European Physical Journal C*, **81**(8), p. 769.
URL `https://doi.org/10.1140/epjc/s10052-021-09555-1`

[117] Bouffanais, Y., M. Mapelli, F. Santoliquido, N. Giacobbo, U. N. Di Carlo, S. Rastello, M. C. Artale, and G. Iorio (2021) “New insights on binary black hole formation channels after GWTC-2: young star clusters versus isolated binaries,” *Monthly Notices of the Royal Astronomical Society*, **507**(4), pp. 5224–5235, `https://academic.oup.com/mnras/article-pdf/507/4/5224/40392757/stab2438.pdf`.
URL `https://doi.org/10.1093/mnras/stab2438`

[118] Collaboration, T. L. S. and the Virgo Collaboration (2019) “Binary Black Hole Population Properties Inferred from the First and Second Observing Runs of Advanced LIGO and Advanced Virgo,” *The Astrophysical Journal Letters*, **882**(2), p. L24.
URL `https://dx.doi.org/10.3847/2041-8213/ab3800`

[119] Hanna, C. (2024), “manifold,” `https://git.ligo.org/chad-hanna/manifold`.

[120] (2022), “LIGO Sensitivity (190 Mpc) (used for O4 simulations),” .
URL `https://dcc.ligo.org/public/0165/T2000012/002/aligo_O4high.txt`

[121] Fong, H. K. Y. (2018) *From simulations to signals: Analyzing gravitational waves from compact binary coalescences*, Ph.D. thesis, Toronto U.
URL `https://tspace.library.utoronto.ca/handle/1807/91831`

[122] Cannon, K., A. Chapman, C. Hanna, D. Keppel, A. C. Searle, and A. J. Weinstein (2010) “Singular value decomposition applied to compact binary coalescence gravitational-wave signals,” *Phys. Rev. D*, **82**, p. 044025.
URL `https://link.aps.org/doi/10.1103/PhysRevD.82.044025`

[123] Buonanno, A., B. R. Iyer, E. Ochsner, Y. Pan, and B. S. Sathyaprakash (2009) “Comparison of post-Newtonian templates for compact binary inspiral signals

in gravitational-wave detectors," *Phys. Rev. D*, **80**, p. 084043.
URL https://link.aps.org/doi/10.1103/PhysRevD.80.084043

[124] Bohé, A., L. Shao, A. Taracchini, A. Buonanno, S. Babak, I. W. Harry, I. Hinder, S. Ossokine, M. Pürrer, V. Raymond, T. Chu, H. Fong, P. Kumar, H. P. Pfeiffer, M. Boyle, D. A. Hemberger, L. E. Kidder, G. Lovelace, M. A. Scheel, and B. Szilágyi (2017) "Improved effective-one-body model of spinning, nonprecessing binary black holes for the era of gravitational-wave astrophysics with advanced detectors," *Phys. Rev. D*, **95**, p. 044028.
URL https://link.aps.org/doi/10.1103/PhysRevD.95.044028

[125] Callister, T., J. Kanner, T. Massinger, S. Dhurandhar, and A. Weinstein (2017) "Observing gravitational waves with a single detector," *Classical and Quantum Gravity*, **34**(15), p. 155007.

[126] Ligo Scientific Collaboration, VIRGO Collaboration, and Kagra Collaboration (2024) "LIGO/Virgo/KAGRA S240422ed: Updated significance estimate," *GRB Coordinates Network*, **36812**, p. 1.

[127] Messick, C. (2019) *Detecting Gravitational Waves for Multi-Messenger Astro*, Ph.D. thesis.

[128] Collaboration, L. S. and V. Collaboration, "GWTC-2 Data Release: Sensitivity of Matched Filter Searches to Binary Black Hole Merger Populations," https://dcc.ligo.org/P2000217/public.

[129] Abbott, R., T. D. Abbott, F. Acernese, K. Ackley, C. Adams, N. Adhikari, R. X. Adhikari, V. B. Adya, C. Affeldt, D. Agarwal, et al. (2022) "Search for intermediate-mass black hole binaries in the third observing run of Advanced LIGO and Advanced Virgo," *Astronomy & astrophysics*, **659**, p. A84.

[130] the LIGO Scientific Collaboration, the Virgo Collaboration, and the KAGRA Collaboration (2023), "Search for gravitational-lensing signatures in the full third observing run of the LIGO-Virgo network," 2304.08393.

[131] Abbott, B. P. et al. (2017) "Multi-messenger Observations of a Binary Neutron Star Merger," *Astrophys. J. Lett.*, **848**, L12, 1710.05833.

[132] Mukherjee, D. et al. (2018) "The GstLAL template bank for spinning compact binary mergers in the second observation run of Advanced LIGO and Virgo," 1812.05121.

[133] "IGWN Public Alerts User Guide:Observing Capabilities," https://emfollow.docs.ligo.org/userguide/capabilities.html.

[134] "Bottle: Python Web Framework: https://bottlepy.org/docs/dev/," .

[135] The LIGO Scientific Collaboration, The Virgo Collaboration, The KAGRA Collaboration, R. Abbott, et al. (2023), "Open data from the third observing run of LIGO, Virgo, KAGRA and GEO," .
URL `https://arxiv.org/abs/2302.03676`

[136] Abbott, B. P. et al. (2020) "GW190425: Observation of a Compact Binary Coalescence with Total Mass $\sim 3.4 M_{\odot}$," `2001.01761`.

[137] Hanna, C., P. Joshi, R. Huxford, K. Cannon, S. Caudill, C. Chan, B. Cousins, J. D. E. Creighton, B. Ewing, M. Fernandez, H. Fong, P. Godwin, R. Magee, D. Meacher, C. Messick, S. Morisaki, D. Mukherjee, H. Ohta, A. Pace, S. Privitera, S. Sachdev, S. Sakon, D. Singh, R. Tapia, L. Tsukada, D. Tsuna, T. Tsutsui, K. Ueno, A. Viets, L. Wade, M. Wade, and J. Wang (2022) "Metric assisted stochastic sampling search for gravitational waves from binary black hole mergers," *Phys. Rev. D*, **106**, p. 084033.
URL `https://link.aps.org/doi/10.1103/PhysRevD.106.084033`

[138] Abbott, B. P. et al. (2016) "GW151226: Observation of Gravitational Waves from a 22-Solar-Mass Binary Black Hole Coalescence," *Phys. Rev. Lett.*, **116**(24), p. 241103, `1606.04855`.

[139] Abbott, R., T. Abbott, S. Abraham, F. Acernese, K. Ackley, A. Adams, C. Adams, R. Adhikari, V. Adya, C. Affeldt, et al. (2021) "Observation of gravitational waves from two neutron star–black hole coalescences," *The Astrophysical Journal Letters*, **915**(1), p. L5.

[140] Trovato, A. (2020) "GWOSC: Gravitational Wave Open Science Center," *PoS*, **Asterics2019**, p. 082.

[141] Zackay, B., L. Dai, T. Venumadhav, J. Roulet, and M. Zaldarriaga (2019) "Detecting Gravitational Waves With Disparate Detector Responses: Two New Binary Black Hole Mergers," `1910.09528`.

[142] Venumadhav, T., B. Zackay, J. Roulet, L. Dai, and M. Zaldarriaga (2019) "New Binary Black Hole Mergers in the Second Observing Run of Advanced LIGO and Advanced Virgo," `1904.07214`.

[143] Zackay, B., T. Venumadhav, L. Dai, J. Roulet, and M. Zaldarriaga (2019) "Highly spinning and aligned binary black hole merger in the Advanced LIGO first observing run," *Phys. Rev.*, **D100**(2), p. 023007, `1902.10331`.

[144] Venumadhav, T., B. Zackay, J. Roulet, L. Dai, and M. Zaldarriaga (2019) "New search pipeline for compact binary mergers: Results for binary black holes in the first observing run of Advanced LIGO," *Phys. Rev.*, **D100**(2), p. 023011, `1902.10341`.

[145] Nitz, A. H., C. D. Capano, S. Kumar, Y.-F. Wang, S. Kastha, M. Schäfer, R. Dhurkunde, and M. Cabero (2021), "3-OGC: Catalog of gravitational waves from compact-binary mergers," 2105.09151.

[146] Finn, L. S. and D. F. Chernoff (1993) "Observing binary inspiral in gravitational radiation: One interferometer," *Phys. Rev.*, **D47**, pp. 2198–2219, gr-qc/9301003.

[147] Owen, B. J. (1996) "Search templates for gravitational waves from inspiraling binaries: Choice of template spacing," *Phys. Rev. D*, **53**, pp. 6749–6761.
URL https://link.aps.org/doi/10.1103/PhysRevD.53.6749

[148] Allen, B., W. G. Anderson, P. R. Brady, D. A. Brown, and J. D. E. Creighton (2012) "FINDCHIRP: An Algorithm for detection of gravitational waves from inspiraling compact binaries," *Phys. Rev.*, **D85**, p. 122006, gr-qc/0509116.

[149] Babak, S., R. Biswas, P. Brady, D. A. Brown, K. Cannon, C. D. Capano, J. H. Clayton, T. Cokelaer, J. D. Creighton, T. Dent, et al. (2013) "Searching for gravitational waves from binary coalescence," *Physical Review D*, **87**(2), p. 024033.

[150] Nitz, A. H., T. Dent, T. Dal Canton, S. Fairhurst, and D. A. Brown (2017) "Detecting binary compact-object mergers with gravitational waves: Understanding and Improving the sensitivity of the PyCBC search," *Astrophys. J.*, **849**(2), p. 118, 1705.01513.

[151] Harry, I. W., B. Allen, and B. Sathyaprakash (2009) "Stochastic template placement algorithm for gravitational wave data analysis," *Physical Review D*, **80**(10), p. 104014.

[152] Ajith, P., N. Fotopoulos, S. Privitera, A. Neunzert, N. Mazumder, and A. Weinstein (2014) "Effectual template bank for the detection of gravitational waves from inspiralling compact binaries with generic spins," *Physical Review D*, **89**(8), p. 084041.

[153] Harry, I., S. Privitera, A. Bohé, and A. Buonanno (2016) "Searching for gravitational waves from compact binaries with precessing spins," *Physical Review D*, **94**(2), p. 024012.

[154] Cornish, N. J. and J. Crowder (2005) "LISA data analysis using Markov chain Monte Carlo methods," *Physical Review D*, **72**(4), p. 043005.

[155] Veitch, J., V. Raymond, B. Farr, W. Farr, P. Graff, S. Vitale, B. Aylott, K. Blackburn, N. Christensen, M. Coughlin, W. Del Pozzo, F. Feroz, J. Gair, C.-J. Haster, V. Kalogera, T. Littenberg, I. Mandel, R. O'Shaughnessy, M. Pitkin, C. Rodriguez, C. Röver, T. Sidery,

R. Smith, M. Van Der Sluys, A. Vecchio, W. Vousden, and L. Wade (2015) "Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library," *Phys.Rev.D*, **91**(4), p. 042003, 1409.7215.

[156] et al. (LIGO Scientific Collaboration, R. A. and V. Collaboration) (2021) "Open data from the first and second observing runs of Advanced LIGO and Advanced Virgo," *SoftwareX*, **13**, p. 100658.
URL https://www.sciencedirect.com/science/article/pii/S2352711021000030

[157] (2023), "GstLAL," https://git.ligo.org/lscsoft/gstlal.

[158] (2021), "Open Science Grid: Introduction," .
URL https://opensciencegrid.org/about/introduction/

[159] LIGO Scientific Collaboration (2018), "LIGO Algorithm Library - LALSuite," free software (GPL).

[160] Ashton, G., S. Thiele, Y. Lecoeuche, J. McIver, and L. K. Nuttall (2021), "Parameterised population models of transient non-Gaussian noise in the LIGO gravitational-wave detectors," 2110.02689.

[161] Davis, D., L. V. White, and P. R. Saulson (2020) "Utilizing aLIGO glitch classifications to validate gravitational-wave candidates," *Classical and Quantum Gravity*, **37**(14), p. 145001.
URL http://dx.doi.org/10.1088/1361-6382/ab91e6

[162] Fairhurst, S. (2009) "Triangulation of gravitational wave sources with a network of detectors," *New Journal of Physics*, **11**(12), p. 123006.

[163] Nitz, A. H., T. Dent, G. S. Davies, S. Kumar, C. D. Capano, I. Harry, S. Mozzon, L. Nuttall, A. Lundgren, and M. Tápai (2019) "2-OGC: Open Gravitational-wave Catalog of binary mergers from analysis of public Advanced LIGO and Virgo data," 1910.05331.

[164] LSC Algorithm Library, "LSC Algorithm Library," http://www.lsc-group.phys.uwm.edu/lal, uRL http://www.lsc-group.phys.uwm.edu/lal.

[165] Aasi, J. et al. (2013) "Parameter estimation for compact binary coalescence signals with the first generation gravitational-wave detector network," *Phys. Rev. D*, **88**, p. 062001.
URL https://link.aps.org/doi/10.1103/PhysRevD.88.062001

[166] (2023), "gwcelery," https://git.ligo.org/emfollow/gwcelery.

[167] Foundation, A. S., "Apache Kafka," .
URL https://kafka.apache.org/

[168] Sidery, T., B. Aylott, N. Christensen, B. Farr, W. Farr, F. Feroz, J. Gair, K. Grover, P. Graff, C. Hanna, V. Kalogera, I. Mandel, R. O'Shaughnessy, M. Pitkin, L. Price, V. Raymond, C. Röver, L. Singer, M. van der Sluys, R. J. E. Smith, A. Vecchio, J. Veitch, and S. Vitale (2014) "Reconstructing the sky location of gravitational-wave detected compact binary systems: Methodology for testing and comparison," *Phys. Rev. D*, **89**, p. 084060.
URL `https://link.aps.org/doi/10.1103/PhysRevD.89.084060`

[169] Newmarch, J. and J. Newmarch (2017) "GStreamer," *Linux Sound Programming*, pp. 211–221.

Vita

Prathamesh Joshi

Education

The Pennsylvania State University 2019–2024
Ph.D. in Physics *(Advisor: Chad Hanna)*

The Indian Institute of Technology, Bombay 2015–2019
B.Tech. in Engineering Physics

Fellowships and Awards

David H. Rank Memorial Physics Award 2020
The Pennsylvania State University

Homer F. Braddock Fellowship 2019, 2020
The Pennsylvania State University

Select Publications

P. Joshi, et al. *How Many Times Should We Matched Filter Gravitational Wave Data? A Comparison of GstLAL's Online and Offline Performance*, 2024.

P. Joshi, et al. *Method to get Better Sky Maps in a GstLAL Low-Latency Analysis*, 2024.

P. Joshi, et al. *New Methods for Offline GstLAL Analyses*, 2024.

P. Joshi, et al. *Method for removing signal contamination during significance estimation of a GstLAL analysis*, *Physical Review D*, vol. 108, p. 084032, Oct. 2023.

L. Tsukada, **P. Joshi**, et al. *Improved ranking statistics of the GstLAL inspiral search for compact binary coalescences*, *Physical Review D*, vol. 108, no. 4, Aug. 2023.

C. Hanna, **P. Joshi**, et al. *Metric assisted stochastic sampling search for gravitational waves from binary black hole mergers*, *Physical Review D*, vol. 106, no. 8, Oct. 2022.

Collaboration publications with major contributions

A. G. Abac, et al. *Observation of Gravitational Waves from the Coalescence of a 2.5-4.5 $M_\odot$ Compact Object and a Neutron Star*, 2024 ApJL 970 L34