Classical and Quantum Gravity

Enhancing the capabilities of LIGO time–frequency plane searches through clustering

To cite this article: R Khan and S Chatterji 2009 *Class. Quantum Grav.* **26** 155009

View the article online for updates and enhancements.

Class. Quantum Grav. **26** (2009) 155009 (14pp) doi:10.1088/0264-9381/26/15/155009

Enhancing the capabilities of LIGO time–frequency plane searches through clustering

R Khan[1] **and S Chatterji**[2]

[1] Columbia Astrophysics Laboratory, Columbia University, Pupin Labs Rm 1027, MC 5247, New York, NY 10027, USA
[2] LIGO Laboratory, California Institute of Technology, MS 18-34, Pasadena, CA 91125, USA

E-mail: rubab@astro.columbia.edu and shourov@ligo.caltech.edu

Received 12 December 2008, in final form 9 June 2009
Published 10 July 2009
Online at stacks.iop.org/CQG/26/155009

Abstract
One class of gravitational wave signals LIGO is searching for consists of short duration bursts of unknown waveforms. Potential sources include core collapse supernovae, gamma ray burst progenitors and mergers of binary black holes or neutron stars. We present a density-based clustering algorithm to improve the performance of time–frequency searches for such gravitational-wave bursts when they are extended in time and/or frequency, and not sufficiently well known to permit matched filtering. We have implemented this algorithm as an extension to the QPipeline, a gravitational-wave data analysis pipeline for the detection of bursts, which currently determines the statistical significance of events based solely on the peak significance observed in minimum uncertainty regions of the time–frequency plane. Density-based clustering improves the performance of such a search by considering the aggregate significance of arbitrarily shaped regions in the time–frequency plane and rejecting the isolated minimum uncertainty features expected from the background detector noise. In this paper, we present test results for simulated signals and demonstrate that density-based clustering improves the performance of the QPipeline for signals extended in time and/or frequency.

PACS numbers: 04.80.Nn, 07.05.Kf, 95.55.Ym, 95.75.Pq

(Some figures in this article are in colour only in the electronic version)

1. Introduction

The first generation of interferometric gravitational wave detectors has now collected data at their design strain sensitivities [1–6], and an improved generation of detectors [7, 8] is already under development. Even at this unprecedented level of sensitivity, potentially detectable signals from astrophysical sources are expected to be at or near the limits of detectability, requiring carefully designed search algorithms in order to identify and distinguish them from the background detector noise. In this study, we focus on the problem of detecting the specific class of gravitational wave signals known as gravitational-wave bursts (GWBs). These are signals lasting from a few milliseconds to a few seconds, for which we do not have sufficient theoretical understanding or reliable models to predict a waveform. This includes signals from the merger of binary compact objects, asymmetric core collapse supernovae, the progenitors of gamma ray bursts and possibly unexpected sources.

Since accurate waveform predictions do not exist for GWBs, the typical method to identify them is to project the data under test onto a convenient basis of abstract waveforms that are chosen to cover a targeted region of the time–frequency plane, and then identify regions of this search space with statistically significant excess signal energy [9]. In this study, we focus on one such burst search algorithm, the QPipeline [10], which first projects the data under test onto an overlapping basis of Gaussian enveloped sinusoids characterized by their center time, center frequency and quality factor. A trigger is recorded whenever this projection exceeds a threshold value, with the magnitude of the projection indicating the significance of the trigger. Since the triggers are considered separately, the existing algorithm currently under-reports the total energy and true significance of those signals that are extended in time and/or frequency, since they have a significant projection onto multiple independent basis functions. Since GWB signals with such extended features are commonly observed in simulations of core collapse supernovae, the mergers of binary compact objects and instabilities of spinning neutron stars, there are good reasons to try to improve the sensitivity of the search algorithm to such sources.

To improve the sensitivity of the QPipeline to signals that are extended in time and/or frequency, we have investigated extensions to the QPipeline that also consider the combined statistical significance of arbitrarily shaped clusters of projections in the time–frequency plane. Although a number of clustering algorithms are commonly available [11], this work focuses on a density-based clustering algorithm due to its ability to also decrease the false detection probability of GWB searches by rejecting isolated single projection events associated with noise fluctuations. In this paper, we present the details of a density-based clustering algorithm implementation as an extension to the QPipeline and demonstrate the resulting improved performance of the QPipeline for signals that are extended in time and/or frequency.

The paper is structured as follows. Section 2 briefly describes the QPipeline burst search algorithm. Section 3 considers the motivations for clustering and surveys some of the available approaches. Section 4 presents the details of the proposed density-based clustering algorithm. Section 5 demonstrates the benefit of the proposed approach for detecting simulated gravitational wave bursts of various waveforms. Finally, in section 6, we present our conclusions and discuss possible future investigations.

2. The QPipeline burst search algorithm

The QPipeline is an analysis pipeline for the detection of GWBs in data from interferometric gravitational wave detectors [10]. It is based on the Q-transform [12], a multi-resolution time–frequency transform that projects the data under test onto the space of Gaussian windowed complex exponentials characterized by a center time τ, center frequency ϕ and quality factor Q

$$X(\tau, \phi, Q) = \int_{-\infty}^{+\infty} x(t)\,\mathrm{e}^{-4\pi^2\phi^2(t-\tau)^2/Q^2}\,\mathrm{e}^{-\mathrm{i}2\pi\phi t}\,\mathrm{d}t \tag{1}$$

The space of Gaussian enveloped complex exponentials is an overlapping basis of waveforms, whose duration σ_t and bandwidth σ_f, defined as the standard deviation of the squared Gaussian envelope in time and frequency, have the minimum possible time–frequency uncertainty, $\sigma_t \sigma_f = 1/4\pi$, where $Q = \phi/\sigma_f$.

There is good reason to select an overlapping basis of multi-resolution minimum-uncertainty functions. In the absence of detailed knowledge of the gravitational waveform, such a basis provides the tightest possible constraints on the time–frequency area of unmodeled signals, permitting the time–frequency distribution of signal energy to be non-coherently reconstructed while incorporating as little noise energy as possible. A choice of basis that does not have minimum time–frequency uncertainty would typically include more noise than necessary, decreasing signal-to-noise ratio. The exception would be a restricted search for a known set of waveforms, in which a matched filter search, where the template matches the target signal, would be optimal. This type of restricted search is not the focus of this paper. Another benefit of the tighter time–frequency constraints afforded by a multi-resolution sine-Gaussian template bank is the decreased likelihood for false coincidences, when testing for coincidence between multiple detectors.

In practice, the Q transform is evaluated only for a finite number of basis functions, also referred to here as templates or tiles. These templates are selected to cover a targeted region of signal space and are spaced such that the fractional signal energy loss $-\delta Z/Z$ due to the mismatch $\delta\tau$, $\delta\phi$ and δQ between an arbitrary basis function and the nearest measurement template,

$$\frac{-\delta Z}{Z} \simeq \frac{4\pi^2\phi^2}{Q^2}\delta\tau^2 + \frac{2+Q^2}{4\phi^2}\delta\phi^2 + \frac{1}{2Q^2}\delta Q^2 - \frac{1}{\phi Q}\delta\phi\delta Q, \tag{2}$$

is no larger than ~20%. This naturally leads to a tiling of the signal space that is logarithmic in Q, logarithmic in frequency and linear in time.

The statistical significance of Q transform projections are given by their normalized energy Z, defined as the ratio of the squared projection magnitude to the mean-squared projection magnitude of other templates with the same central frequency and Q. For the case of ideal white noise, Z is exponentially distributed and related to the matched filter SNR ρ [13] by the relation

$$Z = |X|^2/\langle |X|^2\rangle_\tau = -\ln P(Z' > Z) = \rho^2/2. \tag{3}$$

The QPipeline consists of the following steps. The data are first whitened by zero-phase linear predictive filtering [10, 14]. Next, the Q transform is applied to the whitened data, and normalized energies are computed for each measurement template. Templates with statistically significant signal content are then identified by applying a threshold on the normalized energy. Finally, since a single event may potentially produce multiple overlapping triggers due to the overlap between measurement templates, only the most significant of overlapping templates are reported as triggers. As a result, the QPipeline is effectively a templated matched filter search [13] for signals that are Gaussian enveloped sinusoids in the whitened signal space.

Figure 1 shows an example of the QPipeline applied to the inspiral phase of a simulated binary neutron star coalescence signal injected into typical single detector LIGO data with an SNR of 48.2.

3. Motivations and options for clustering

Currently, the QPipeline considers the significance of triggers independently. The detectability of a particular GWB signal therefore depends upon its maximum single projection onto the

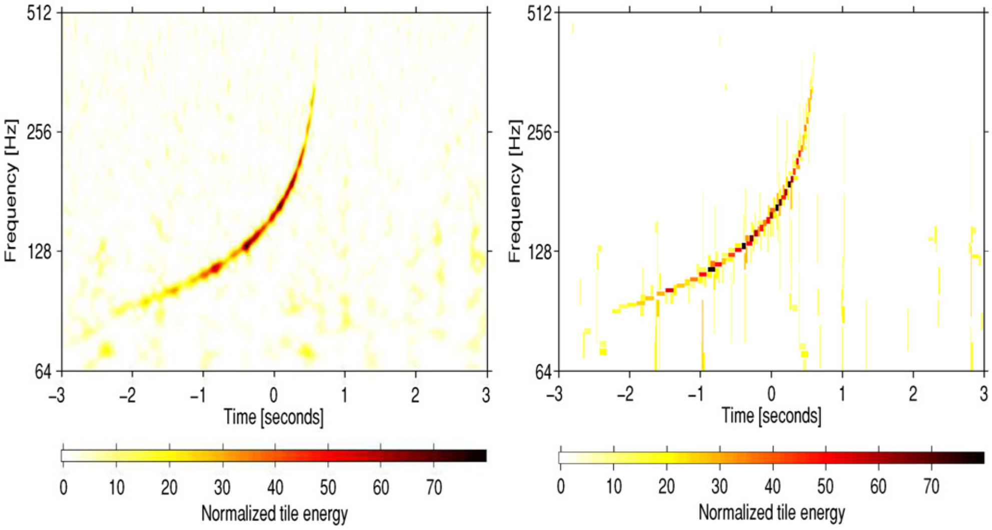

Figure 1. The QPipeline view of the inspiral phase of a simulated optimally oriented 1.4/1.4 solar mass binary neutron star merger injected into typical single detector LIGO data with an SNR of 48.2 as measured by a matched filter search targeting inspiral signal. The QPipeline projects the whitened data onto the space of time, frequency, and Q. The *left panel* image shows the resulting time–frequency spectrogram of normalized signal energy for the value of Q that maximizes the measured normalized energy, while the *right panel* image shows the time–frequency distribution of only the most significant non-overlapping triggers regardless of Q.

space of Gaussian enveloped sinusoids. As a result, the QPipeline is most sensitive to signals with near minimum time–frequency uncertainty and less sensitive to signals that are extended in time and/or frequency such that their energy is spread across multiple non-overlapping basis functions. For example, the detectability of the simulated inspiral signal shown in figure 1 is currently determined by the single most significant tile near the center of the signal, which has a single tile SNR of 12.7. This is significantly less than the SNR of 48.2 that is recovered by a matched filter search tuned for this waveform.

Although the above example focuses on binary neutron star inspiral waveforms, matched filter search methods are more appropriate to search for such signals, since there waveform is well understood [15–18]. We focus on them here only because they represent an astrophysically interesting example case of a signal which is extended in time and frequency. The QPipeline is not intended to search for inspiral signals, but instead focuses on the search for other transient sources such as the less well-understood merger phase of coalescing binary compact objects, core collapse supernovae and instabilities in spinning neutron stars, many of which are also expected to produce waveforms that are extended in time and/or frequency [19–22]. As a result, we seek a method to improve the sensitivity of the QPipeline to signals that are extended in time and/or frequency that is applicable to the general case of astrophysically unmodeled bursts and is not specific to any one particular waveform.

An obvious solution is to simultaneously consider the aggregate significance of all tiles that comprise the signal. This requires an approach that identifies clusters of related tiles in the time–frequency plane. In this context, we define clustering as the grouping of the set of all significant QPipeline tiles into subsets, such that all tiles within a subset are closely related by their relative distance in the time–frequency plane.

There are many different clustering methods available; however, they all tend to fall into three categories [11].

Partitioning methods. The classical example of a partitioning method is the K-means algorithm [24]. In K-means clustering, a fixed number of clusters is presupposed and an initial guess to partition objects into these clusters are made. A centroid is computed for each cluster, and the total sum distance of all objects from their cluster centroid is computed as a figure of merit. K-means iteratively reassigns objects to different clusters until this figure of merit is minimized.

There are a number of drawbacks to the K-means approach, as described in [11, 24]. The first is that it presupposes a fixed number of clusters, though there are variations that allow the number of clusters to change [25]. The other difficulty with the K-means approach is the tendency to produce spherical or ellipsoidal clusters rather than more complicated arbitrary shapes. This is acceptable for some signal morphologies but not so in general. For example, the signal expected from inspiralling binary compact objects, which is long and extended in time and frequency, would not be easily identified by K-means clustering [25]. A third difficulty with the K-means approach is the sensitivity to the initial guess and the possibility of the algorithm to identify a local rather than global minimum [11].

Hierarchical methods. This type of clustering algorithm first evaluates the pairwise difference between all N objects, then arranges them into a tree structure, where each object to be clustered is a leaf [26]. In the agglomerative hierarchical approach, the tree is constructed in N levels. At each level, the closest pair of leaves and/or branches is merged, with the Nth level representing a cluster of all objects. A cluster is formed by cutting this tree based on some criteria such as a maximum distance or the inconsistency between the distance between cluster leaves or branches and the distance to the next closest leaf or branch.

Hierarchical clustering has the flexibility of producing arbitrary numbers of clusters with arbitrary shape, and is therefore more applicable to the problem of clustering in time, frequency and Q for gravitational-wave burst detection.

The left panel of figure 2 shows an example of hierarchical clustering, implemented using the functions ‘linkage’ and ‘cluster’ from MATLAB Statistics Toolbox[3] applied to the detection of a simulated binary neutron star inspiral signal. The drawback to this approach is that it presupposes that every data point must be included in one cluster or another, even if a cluster is to be constituted of only one data point. As a result, it produces a large number of insignificant clusters and tends to build clusters of unrelated data points.

Density-based methods. Density-based clustering [23] is a variation on the hierarchical clustering approach. Instead of constructing a tree structure, density-based clustering starts with single object and iteratively adds objects to that cluster using a predefined set of criteria based on the density of objects within a given neighborhood radius, until the criterion is no longer met and all objects have been tested.

Like other hierarchical methods, density-based methods also allow an arbitrary number of clusters as well as arbitrary cluster shape. They also have the advantage of rejecting single isolated data points that are not potentially related to a large number of points. Thus, this method only produces clusters with multiple data points and can successfully exclude unrelated points from a cluster. For this reason, we have focused on density-based methods in this work.

[3] Statistics Toolbox Software Version 5.2 (R2006a), http://www.mathworks.com/access/helpdesk/help/toolbox/stats/rn/bqmfe_z-10.html

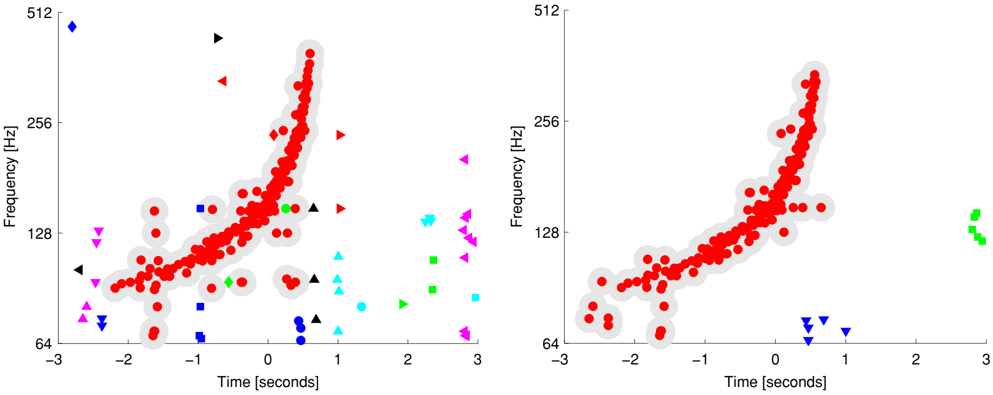

Figure 2. Two different clustering methods were applied to the QPipeline triggers from figure 1. The *left panel* image shows the result of applying a hierarchical-based clustering method [11], while the *right panel* image shows the result of applying the proposed density-based clustering method [23]. Here each combination of color and shape denotes a different cluster. Although both hierarchical and density-based clustering approaches succeed in isolating most of the signal energy within a single cluster, the density-based approach has the additional advantage of discarding isolated triggers due to the background detector noise.

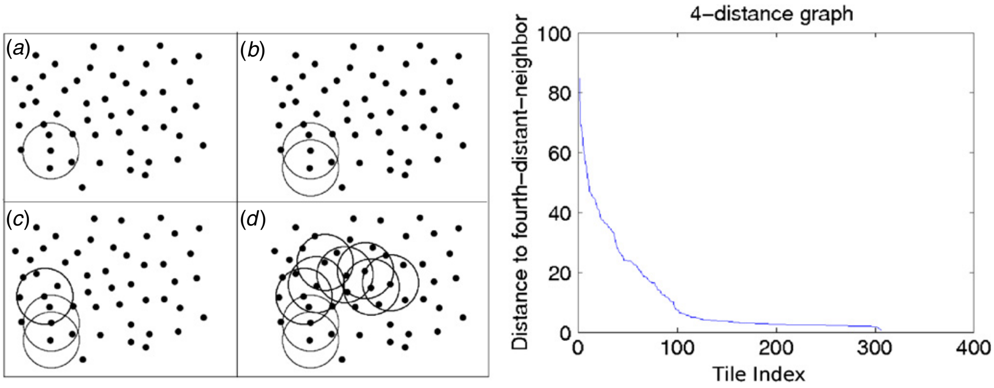

Figure 3. Building clusters from data points using the density-based clustering algorithm, as discussed in details in section 4. The *left panel* shows the steps of building a cluster using density-based clustering. The *right panel* shows the four-distance graph which helps us determine the neighborhood radius. (Figures derived from concepts presented in [23].)

4. Density-based clustering algorithm

Density-based clustering [23] facilitates searches for signals of unknown shape. It does not clutter the output with a list of numerous noise-related clusters that contain just a few significant data points. The algorithm looks for neighbors of those points that have at least a given number of neighboring points within a given distance on the time–frequency plane, and forms clusters of data-points that can be related through their common neighbors. Our implementation of density-based clustering algorithm takes two parameters: minimum neighbor number and neighborhood radius, and it considers each tile produced by QPipeline as a data point.

Density-based clustering first finds tile's nearest neighbors, then that neighbors' neighbors, and so on. The left panel of figure 3(*a*) shows data points before clustering. If the density of data points within a given distance around a point is above a given threshold to form a cluster, that point becomes a cluster seed (*b*). Neighboring data points having a sufficient number of neighbors are then included in the cluster (*c*). This process repeats as long as data points with sufficient number of neighbors are found (*d*).

Any clustering algorithm requires measurement of the pairwise distances between all data points; in our case, the pairwise distance between all tiles is produced by QPipeline. Unfortunately, the tiles have varied shapes, which makes measurement of distance between any pair of tiles somewhat difficult. We have implemented a simple distance metric that addresses the issue of varied tile shapes. For a pair of tiles with center times t_1 and t_2, center frequencies f_1 and f_2, and normalized energy of z_1 and z_2, the distance on the time–frequency plane D is measured from the following relations:

$$D = \sqrt{{D_t}^2 + \beta {D_f}^2} \tag{4}$$

$$D_t = \frac{|t_2 - t_1|}{\overline{\Delta t}}, \qquad D_f = \frac{|f_2 - f_1|}{\overline{\Delta f}}, \tag{5}$$

$$\overline{\Delta t} = \frac{\Delta t_1 z_1 + \Delta t_2 z_2}{z_1 + z_2}, \qquad \overline{\Delta f} = \frac{\Delta f_1 z_1 + \Delta f_2 z_2}{z_1 + z_2}, \tag{6}$$

where D_t is the distance along the timescale, D_f is the distance along the frequency scale, $\overline{\Delta t}$ is the scale factor on the timescale, $\overline{\Delta f}$ is the scale factor on the frequency scale, Δt_1 and Δt_2 are durations, and Δf_1 and Δf_2 are bandwidths. The parameter β is a tuning parameter that determines the relative importance of distance in frequency versus distance in time. It was determined empirically that when $\beta = 30$, the best results were achieved for the extended signals considered in section 5 as measured by the receiver operating characteristic (ROC) curves presented there.

The mismatch metric in equation (2) can also be used [10] for this purpose. However, in this initial study we have chosen to use the simpler approach of equations (4)–(6), and leave the possibility of using equation (2) for future study.

The minimum neighbor number determines which tiles are to be considered as potential cluster seeds, and which ones are to be excluded from the clustering process entirely because they do not have a sufficient number of neighboring tiles. A minimum neighbor number that is too low may result in too many clustering seeds, potentially producing a large number of clusters, whereas a minimum neighbor number that is too high may result in the exclusion of too many tiles.

The neighborhood radius indicates the distance the algorithm searches from a tile in order to find neighboring tiles, and therefore determines the maximum gap over which the algorithm can cluster. A neighborhood radius that is too low may result in a large number of small clusters, whereas a neighborhood radius that is too high may result in the creation of one large cluster consisting of all significant tiles.

The exact numerical value of the neighborhood radius depends on the minimum neighbor number and the distance metric used. We have experimented by varying the minimum neighbor number from 1 to 8 and found that the best results were achieved with a minimum neighbor number of 4, as measured by the ROC curves in section 5. The value of the neighborhood radius was then determined using the four-distance graph in the right panel of figure 3. The four-distance graph shows the number of tiles on the *x*-axis whose fourth closest neighbor is less than the distance shown on the *y*-axis. Motivated by the four-distance plot, we experimented by varying the neighborhood radius between 3 and 12 and found that a value of 8 produced the best results based on the ROC curves in section 5.

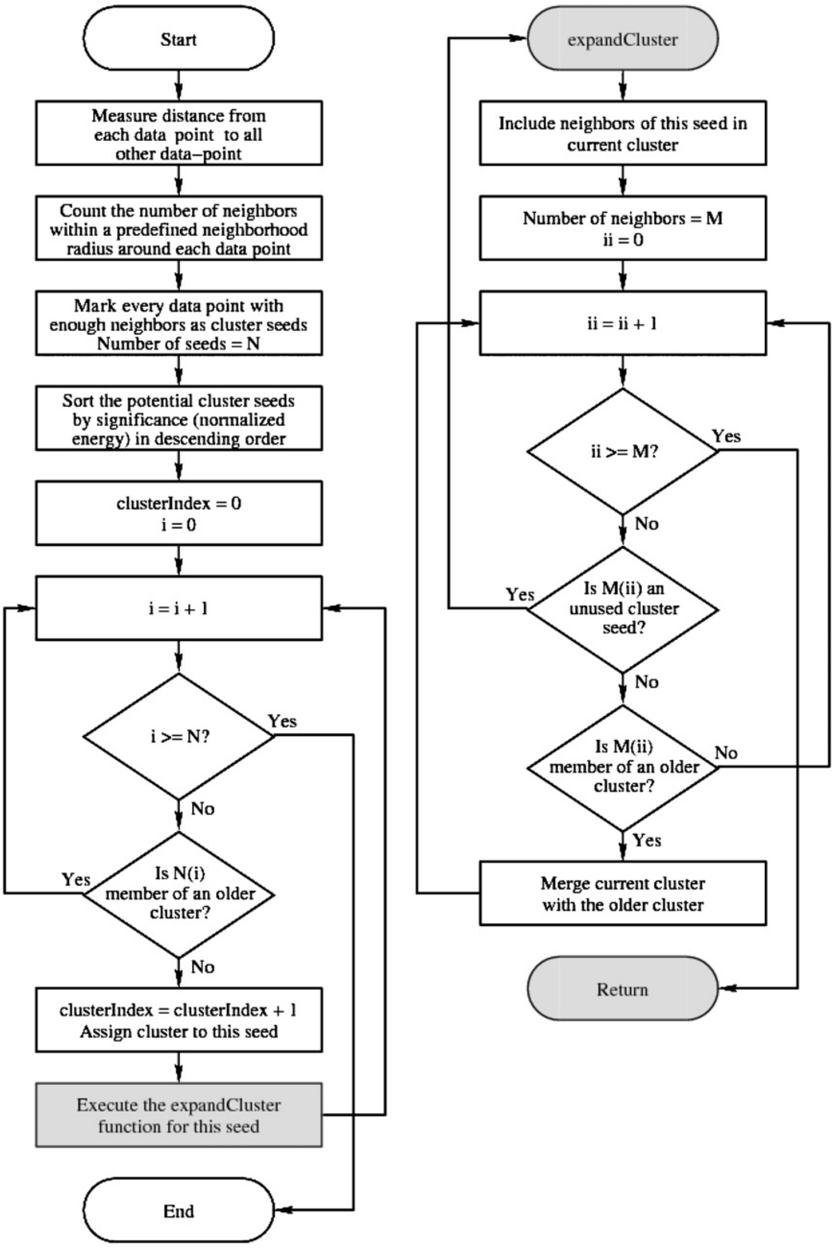
Start
Measure distance from each data point to all other data–point
Count the number of neighbors within a predefined neighborhood radius around each data point
Mark every data point with enough neighbors as cluster seeds Number of seeds = N
Sort the potential cluster seeds by significance (normalized energy) in descending order
clusterIndex = 0
i = 0
i = i + 1
i >= N?
Yes
No
Is N(i) member of an older cluster?
Yes
No
clusterIndex = clusterIndex + 1
Assign cluster to this seed
Execute the expandCluster function for this seed
End
expandCluster
Include neighbors of this seed in current cluster
Number of neighbors = M
ii = 0
ii = ii + 1
ii >= M?
Yes
No
Is M(ii) an unused cluster seed?
Yes
No
Is M(ii) member of an older cluster?
No
Yes
Merge current cluster with the older cluster
Return

Figure 4. Flowchart of density-based clustering algorithm.

A flowchart of the density-based clustering algorithm is shown in figure 4. The main clustering function first uses the distance function to measure pairwise distance between all tiles, and then calls the expandCluster function, which recursively calls itself, to incorporate more tiles into each cluster. The algorithm as presented here was implemented in MATLAB[4].

[4] MATLAB Version 7.2.0.283 (R2006a), http://www.mathworks.com/access/helpdesk/help/techdoc/rn/f0-68730.html

Clustering starts with the highest energy tile that has a sufficient number of neighbors and then proceeds to the next significant tile that also has a sufficient number of neighbors, and is not already assigned to a cluster. If any qualifying member of the current cluster is found to already belong to another cluster, the two clusters are merged. Thus, regardless of which tile the algorithm starts clustering from, it will always find the same clusters for a given set of tiles. For speed optimization, though, our density-based clustering function starts with the more significant tiles first.

The computational cost of the resulting algorithm is dominated by the N^2 cost of computing the distance between all pairs of tiles, where N is the number of tiles. In practice, clustering is applied separately to the $\sim$1 min data blocks used by the QPipeline analysis. This is more than sufficient to detect clusters up to a few seconds in duration, the typical limit of gravitational wave burst searches. At the typical $\sim$1 Hz single detector false rate, the resulting computational cost due to clustering is small compared to that of the rest of the search.

The right panel of figure 2 shows an example cluster constructed using the proposed density-based clustering algorithm. It can be seen that the algorithm has clustered together the most significant part of the previously discussed injection successfully. In addition, almost all of the spurious noise tiles have also been removed. While the high-frequency end of the signal has been lost that part contains very little energy and does not significantly contribute to the significance of the detected trigger.

5. Evaluating performance improvements

We have evaluated our implementation of density-based clustering by measuring its effect on the detection of simulated signals injected into typical single detector LIGO data and its effect on the rate of false detections.

In order to evaluate the false detection rate, the QPipeline was first applied to single detector data without injected signals. This was performed both with and without clustering. Since detectable GWB events are expected to be extremely rare in the few hours of data considered here, and since we have set our thresholds to yield event rates up to $\sim$1 Hz and not demanded coincidence between multiple detectors to reject false events, we can safely identify false events as those events in a given data stretch whose total normalized energy exceeded a specified detection threshold. Three sets of false events were identified: unclustered events, clustered events and combined events formed by the union of unclustered and clustered events. The resulting false event rates as a function of detection threshold are shown in figure 5.

An adverse effect of using density-based clustering is the occasional rejection of highly localized signals, regardless of the detection threshold. This is due to the tendency of density-based clustering to exclude isolated triggers. This is also evident in figure 6, where detection efficiency of sinusoidal Gaussians does not converge to 100% for the case of clustered triggers. To overcome this, we have also considered the performance of a search consisting of the union of both clustered and unclustered triggers and compared it with that for clustered triggers and unclustered triggers only. The resulting sinusoidal Gaussian combined detection efficiency for a given energy threshold is then comparable to that of the unclustered case, as shown in figure 6.

Another possible solution to this problem is to reduce the required number of tiles within the neighborhood radius to zero, permitting single tile clusters. Classical hierarchical clustering also provides an alternative to density-based clustering that permits single tile clusters. Since the focus of this paper is on improved performance for signals that are extended in time and/or frequency, we have not considered either of these alternative choices here.

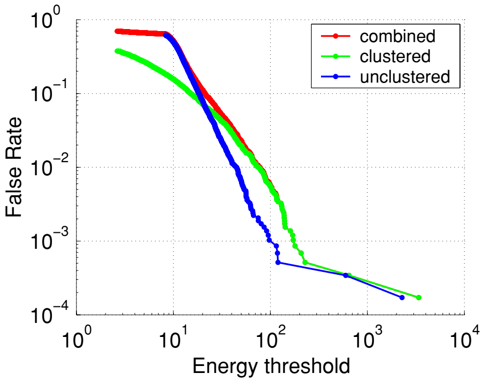

Figure 5. The false event rate of the search algorithm as a function of detection threshold when applied to typical LIGO data. The trigger rate is shown for three different trigger sets: unclustered triggers, clustered triggers, and the union of clustered and unclustered triggers.

The lower false event rate observed in figure 5 for clustered triggers at low detection thresholds is associated with the rejection of isolated noise events as described in section 4. At high detection thresholds, the opposite is true. The presence of transient non-stationary 'glitches' in the data that are extended in time and/or frequency causes the false event rate of clustered triggers to exceed that of unclustered triggers.

To evaluate the effect of clustering on the detection of signals, we next applied the QPipeline to the recovery of simulated signals injected into the same single detector data. Again, this was performed both with and without clustering. Injections were identified as detected if an event was observed above the detection threshold within 1 s of the time of the injected signal. We define the detection efficiency as the fraction of injection signals that were correctly detected and evaluate this efficiency as a function of detection threshold for signals injected at a constant signal-to-noise ratio.

In order to characterize the performance of density-based clustering for a variety of signal morphologies, we have repeated this analysis for five different waveform families. They include simple Gaussian pulses, sinusoidal Gaussian pulses and the fundamental ring down mode of perturbed black holes, which represent signals that are highly localized in the time–frequency plane; and the inspiral phase of coalescing binary compact objects and band-limited time-windowed white noise bursts, which are both extended in the time–frequency plane. Within each waveform family, signals were injected with random parameters such as time, frequency, duration, bandwidth, mass, etc.

Among non-localized signals, inspirals and white noise bursts represent two extremes: white noise bursts fill a large time–frequency region, whereas inspirals, while extended in time and frequency, still only occupy a small portion of a time–frequency region. Three of the waveform families are ad hoc: simple Gaussian pulses, sinusoidal Gaussian pulses and white noise bursts. Two of the waveform families were astrophysical: inspirals and ringdowns. While we are not designing a search to only target known waveforms such as ringdowns and inspirals, they are nonetheless also a useful test case because they are astrophysically motivated and because they can form a basis for comparison with other existing searches, including matched filter searches.

On the left side panels of figure 6, we present the resulting detection efficiency as a function of detection threshold for three of the waveform families that we have considered, representing both ad hoc and astrophysical, as well as localized and extended. On the right

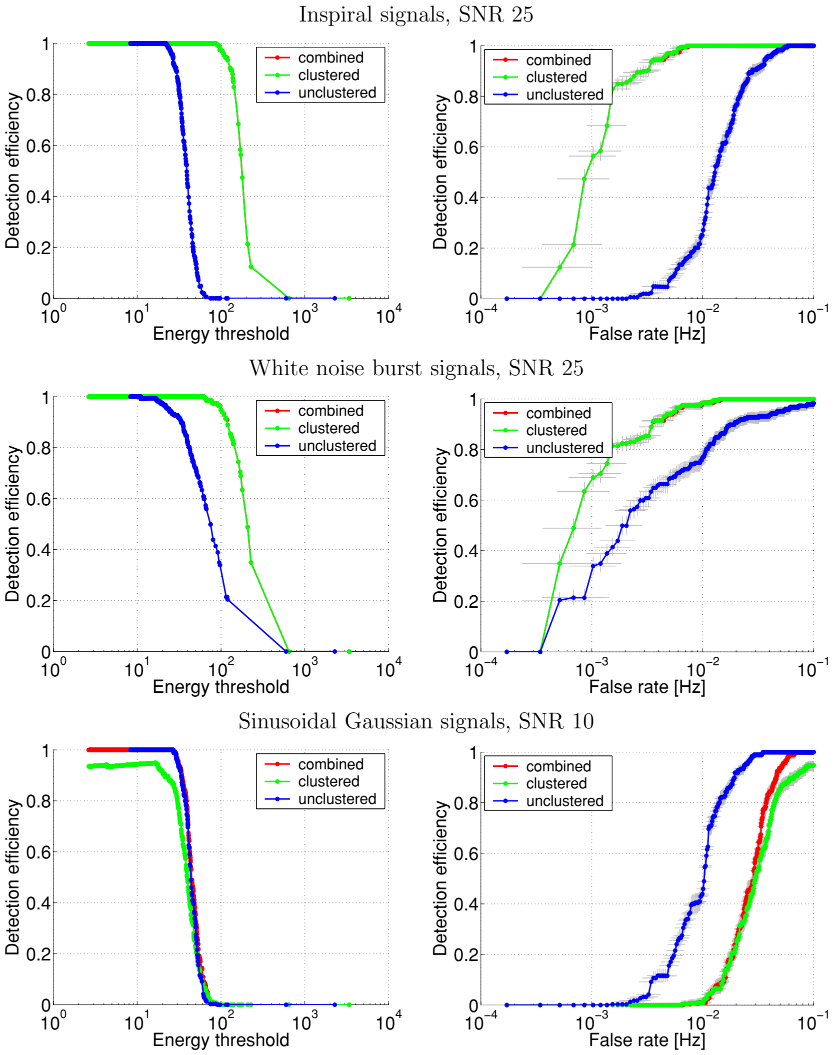

Figure 6. Comparison of the detection efficiency versus search threshold (left) and receiver operator characteristics (right) of the search algorithm, with and without clustering, applied to the detection of simulated inspiral (top), white noise burst (middle) and sinusoidal Gaussian (bottom) waveforms injected 200 times into typical LIGO data at fixed SNR.

side panels of figure 6, we report the receiver operator characteristic for each waveform, which combines the measured false rate from figure 5 with the detection efficiencies from the left side panels.

The results indicate that for the extended waveforms, such as the inspiral and noise burst waveforms, clustering increases search efficiency and significantly improves the resulting ROC by approximately an order of magnitude in false rate. The primary reason for this improved performance is the increase in measured signal energy due to clustering, which is evident as increased detection efficiency in the left-hand side of figure 6.

Although clustering provides a marked improvement for the detection of signals that are extended in time and frequency, figure 6 indicates that clustering also adversely impacts the performance of the search for localized waveforms. In particular, the ROC for sinusoidal Gaussians is worse by roughly a factor of 3 in false rate due to the addition of clustering. The primary cause of this decreased performance is the higher false event rate, which is due to the increased significance of detector glitches after clustering and is evident in figure 5. For signals that are extended in time and/or frequency this higher false event rate is more than compensated by the significant improvement in detection efficiency, but for more localized signals there is no improvement in detection efficiency to compensate for the increased false rate. In practice, we expect the presence of such detector glitches to be largely mitigated by the requirement of a coincident and consistent observation of a gravitational wave in multiple detectors, as well as the absence of a signal in environmental monitors. As a result, the decreased performance for localized signals may also be somewhat mitigated.

6. Conclusions

A density-based method for clustering the measurements from neighboring or overlapping basis functions has been employed to more efficiently detect GWB signals that are extended in time and/or frequency and not well represented by QPipeline's particular choice of basis. The method is capable of identifying an arbitrary number of clusters of arbitrary shape and size, while also rejecting spurious noise triggers, and does not significantly increase the computational cost of the overall QPipeline search.

The proposed clustering algorithm itself is not specific to the QPipeline. Similar improvements are expected when applied to other time–frequency searches for gravitational wave bursts. In particular, the algorithm described here has already been applied to the search for bursts from the soft gamma repeaters using the flare pipeline [27–29]. For estimating upper limits, the flare pipeline initially performed a simple sum over all frequency bins to measure the total signal energy of a trigger. The use of density-based clustering instead improved the flare pipeline's upper limit estimate for 100 ms long white noise bursts in the frequency band from 64 to 1024 Hz by 42%. No improvement was observed for 22 ms long white noise bursts in the band from 100 to 200 Hz, but such signals are fairly localized in the time–frequency plane. These results are consistent with our conclusion that density-based clustering is only beneficial when searching for extended signals.

Our implementation of density-based clustering is already implemented as part of the QPipeline, which has now been incorporated into the Ω-Pipeline [30] for use in future GWB searches.

A number of issues remain open for future investigation. This paper has focused only on single detector data. We have left a study of the effect of density-based clustering on multi-detector GWB searches as a subject for future investigation. In equation (4), we have proposed one possible distance metric. A study of other distance metrics, in particular, ones based on the mismatch metric of equation (2) is also possible. A more in-depth study of hierarchical clustering methods and comparison with the proposed density-based method, as well as previously proposed methods [31–33] is also recommended. Finally, the application of clustering to astrophysical parameter estimation in the event of a detection also warrants further investigation.

Acknowledgments

The authors are grateful for the support of the United States National Science Foundation under cooperative agreement PHY-04-57528, California Institute of Technology, and Columbia

University in the city of New York. We are grateful to the LIGO Scientific Collaboration for their support. We are indebted to many of our colleagues for frequent and fruitful discussion. In particular, we would like to thank Albert Lazzarini for his valuable suggestions regarding this project, and Luca Matone, Zsuzsa Márka, Sharmila Kamat, Jameson Rollins, Peter Kalmus, John Dwyer, Patrick Sutton, Eirini Messeritaki and Szabolcs Márka for their thoughtful comments on the paper. The authors gratefully acknowledge the LIGO Scientific Collaboration hardware injection team for providing the data used in figures 1 and 2. We gratefully acknowledge the contributions of all the software developers and programmers in the broader scientific community without whose incremental achievements over many decades we would not be able to reach this point where implementing this project has become possible. The authors gratefully acknowledge the support of the United States National Science Foundation for the construction and operation of the LIGO Laboratory and the Particle Physics and Astronomy Research Council of the United Kingdom, the Max-Planck-Society and the State of Niedersachsen/Germany for support of the construction and operation of the GEO600 detector. The authors also gratefully acknowledge the support of the research by these agencies and by the Australian Research Council, the Natural Sciences and Engineering Research Council of Canada,the Council of Scientific and Industrial Research of India, the Department of Science and Technology of India, the Spanish Ministerio de Educaciony Ciencia, The National Aeronautics and Space Administration, the John Simon Guggenheim Foundation, the Alexander von Humboldt Foundation,the Leverhulme Trust, the David and Lucile Packard Foundation, the Research Corporation, and the Alfred P Sloan Foundation. The LIGO Observatories were constructed by the California Institute of Technology and Massachusetts Institute of Technology with funding from the National Science Foundation under cooperative agreement PHY-9210038. The LIGO Laboratory operates under cooperative agreement PHY-0107417. This document has been assigned LIGO document number LIGO-P070041-01-Z.

References

[1] Sigg D and (the LIGO Scientific Collaboration 2008 Status of the LIGO detectors *Class. Quantum Grav.* **25** 114041

[2] LIGO Scientific Collaboration 2007 LIGO detector noise curves during the S5 run http://www.ligo.caltech.edu/docs/G/G060009-03/

[3] Acernese F *et al* 2006 *Class. Quantum Grav.* **23** S63

[4] Lück H *et al* 2006 *Class. Quantum Grav.* **23** S71

[5] Beauville F *et al* 2006 Benefits of joint LIGO–Virgo coincidence searches for burst and inspiral signals *J. Phys. Conf. Ser.* **32** 212–22

[6] Abbott B *et al* 2004 Detector description and performance for the first coincidence observations between LIGO and GEO *Nucl. Instrum. Methods Phys. Res.* A **517** 154–79

[7] Adhikari S, Waldman R and Fritschel P (Enhanced LIGO) 2006 *LIGO Technical Document T060156*

[8] http://www.astro.cf.ac.uk/geo/advligo/

[9] Anderson W G, Brady P R, Creighton J D E and Flanagan E E 2001 An excess power statistic for detection of burst sources of gravitational radiation *Phys. Rev.* D **63** 042003

[10] Chatterji S K 2005 The search for gravitational wave bursts in data from the second LIGO science run *PhD Thesis* Massachusetts Institute of Technology

[11] Jain A K, Murty M N and Flynn P J 1999 Data clustering: a review *ACM Comput. Surv.* **31** 264–323

[12] Chatterji S, Blackburn L, Martin G and Katsavounidis E 2004 Multiresolution techniques for the detection of gravitational-wave bursts *Class. Quantum Grav.* **21** S1809–S1818

[13] Dewey D 1986 Analysis of gravity wave antenna signals with matched filter techniques *Proc. Fourth Marcel Grossmann Meeting on General Relativity (Rome, Italy, 1985)* p 581–90

[14] Makhoul J 1975 Linear prediction: a tutorial review *Proc. IEEE* **63** 561–80

[15] van der Sluys M V, Röver C, Stroeer A, Raymond V, Mandel I, Christensen N, Kalogera V, Meyer R and Vecchio A 2008 Gravitational-wave astronomy with inspiral signals of spinning compact-object binaries *Astrophys. Lett.* **688** L61–L64

[16] van der Sluys M, Raymond V, Mandel I, Röver C, Christensen N, Kalogera V, Meyer R and Vecchio A 2008 Parameter estimation of spinning binary inspirals using Markov Chain Monte Carlo *Class. Quantum Grav.* **25** 184011

[17] Gouaty R and (for the LIGO Scientific Collaboration) 2008 Detection confidence tests for burst and inspiral candidate events *Class. Quantum Grav.* **25** 184006

[18] Abbott B *et al* 2008 Search of S3 LIGO data for gravitational wave signals from spinning black hole and neutron star binary inspirals *Phys. Rev.* D **78** 042002

[19] Baumgarte T, Brady P R, Creighton J D E, Lehner L, Pretorius F and Devoe R 2008 Learning about compact binary merger: the interplay between numerical relativity and gravitational-wave astronomy *Phys. Rev.* D **77** 084009

[20] Summerscales T Z, Burrows A, Finn L S and Ott C D 2008 Maximum entropy for gravitational wave data analysis: inferring the physical parameters of core-collapse supernovae *Astrophys. J.* **678** 1142–57

[21] Ott C D 2009 The gravitational wave signature of core-collapse supernovae *Class. Quantum Grav.* **26** 063001

[22] Dimmelmeier H, Ott C D, Marek A and Janka H-T 2008 Gravitational wave burst signal from core collapse of rotating stars *Phys. Rev.* D **78** 064056

[23] Ester M *et al* 1996 A density-based algorithm for discovering clusters in large spatial databases with noise *KDD-96 Proc.* pp 226–31

[24] Kanungo T *et al* 2002 An efficient k-means clustering algorithm: analysis and implementation *IEEE Trans. Pattern Anal. Mach. Intell.* **24** 881–92

[25] Tang L R, Lei H, Mukherjee S and Mohanty S 2008 Cluster analysis of simulated gravitational wave triggers using s-means and constrained validation clustering *Class. Quantum Grav.* **25** 184023

[26] Krishnamachari S and Abdel-Mottaleb M 1998 Hierarchical clustering algorithm for fast image retrieval *Proc. SPIE (Storage and Retrieval for Image and Video Databases VII* vol 3656*)* ed M M Yeung, B-L Yeo and C A Bouman pp 427–35

[27] Kalmus P *et al* 2007 Search method for unmodeled transient gravitational waves associated with SGR flares *Class. Quantum Grav.* **24** S659–S669

[28] Abbott B *et al* 2008 Search for gravitational-wave bursts from soft gamma repeaters *Phys. Rev.* L **101** 211102

[29] Abbott B *et al* 2009 Stacked search for gravitational waves from the 2006 SGR 1900+14 Storm *LIGO Document Control Center* P090024

[30] Abbott B *et al* 2009 Search for gravitational-wave bursts in the first year of the fifth LIGO science run *LIGO Document Control Center* P080056

[31] Sylvestre J 2002 Time–frequency detection algorithm for gravitational wave bursts *Phys. Rev.* D **66** 102004

[32] Honda R *et al* 2008 Astrophysically motivated time–frequency clustering for burst gravitational wave search: application to TAMA300 data *Class. Quantum Grav.* **25** 184035

[33] Klimenko S, Yakushin I, Rakhmanov M and Mitselmakher G 2004 Performance of the waveburst algorithm on LIGO data *Class. Quantum Grav.* **21** 1685