Simulations of Gravitational Wave Generating Transient Objects on Telescope Images

Greg Dooley, Princeton University.
Under the guidance of Eric Chassande-Mottin, University of Paris 7.

Introduction

Detectable gravitational waves (GWs) are expected to arise from massive astrophysical events. In addition to GWs, these events produce powerful electromagnetic emissions whose observation is the primary goal of the LOOC UP project. An acronym for "Locating and Observing Optical Counterparts to Unmodeled Pulses in GW," LOOC UP is a subgroup of the LIGO and VIRGO collaboration [citation]. (footnote to their website: https://geco.phys.columbia.edu/projects/loocup/wiki). Currently, if the three ground based interferometer observatories made a GW detection, limited knowledge could be acquired about its progenitor. With LOOC UP fully developed, robotic telescopes could automatically search the correct portion of the sky for the GW source and provide crucial new information on the supposed object.

When any GW signal above a certain signal to noise ratio threshold is detected, the LOOC UP group's software, LUMIN, should begin analysis. The code will first localize a region or set of regions in the sky where the GW signal most likely came from. It will then decides upon an appropriate observation schedule and choose a telescope to make the observations [citation]. The robotic telescopes TAROT, ROTSE and SkyMapper are among possible collaborators. After recording data in the optical band, the telescope images must be processed to search for transient objects such as supernova and gamma ray bursts (GRBs) which are the most probable GW progenitors.

Whether an object is found or not, this procedure can greatly aid the GW search. [footnote to more reasons for this search]Finding an object in the right portion of the sky and identifying it as one that can emit GWs will help tremendously in convincing the scientific community that a GW was detected. It increases the probability that the trigger was a real event rather than background noise generated on Earth. With improved positive detection abilities, the detectors effectively become more sensitive. They can identify weaker GW signals, lower signal to noise ratios, and still treat them as possible events, scanning the sky for the source. This allows for searches to greater distances, increasing the chance of an actual detection. LIGO and VIRGO would gain this sensitivity by triggering the LUMIN software pipeline as much as several times per week. [kanner]

However, this benefit can only be achieved with reliable automated software to look for transient objects. Out of necessity, all aspects of LUMIN need to be rapid and automatic up through the telescope observation. A short lived optical afterglow of an event could easily be missed if manual decision making slowed down the process too much. On the other hand, the image processing could potentially be investigated manually after the observations. In this way, strong GW candidates could still be correlated to astronomical images with the potential to uncover a convincing astrophysical source. While manual image processing is reliable, it is very slow.[source about Sloan]. To make telescope observations several times per week requires automated transient object identification software. Additionally, real time data analysis could enable a narrow field telescope to make more localized observations of an interesting object. This final step of the LOOC UP project is currently the least developed and the subject of this paper.

Image Processing Software

Many considerations need to go into developing the image processing software. The basic strategy is to analyze a sequence of telescope images for transient objects that match some expected light curve. All telescope images are assumed to have been pre-processed. That is, all bias subtractions, flat fielding, dark current corrections etc. have been made. These correct for noise introduced by the telescope itself [citation]. Additionally, standard photometric calibration is expected to be performed. This sets the zero point magnitude for the image so that the brightness of pixels can be converted into an apparent magnitude. What is left is an image of the sky as shown in figure 1.

With a set of these images, several transient object identification strategies are possible. One of the collaborating telescopes, TAROT, uses the following simplified strategy. Every object on each image is catalogued by running a piece of freeware called SExtractor on it. Stars in each catalogue are matched up to each other and their magnitudes are compared across images. The light curves for transient objects are then produced and the object is identified based on this light curve. [TAROT paper]

While this strategy works for TAROT, large scale surveys must use more sophisticated techniques to avoid high false alarm rates. False alarms can be produced by "flaring stars, comets, asteroids, meteorites, satellites, airplanes, hot pixels, and image defects"[RAPTOR]. The RAPTOR telescope has its own automated real time data analysis to search for transient objects associated with GRB afterglows. Similarly, the Sloan Digital Sky Survey II has its own image processing algorithm to identify supernova [obj classification paper].

The exact nature of the image processing must be specifically tailored to the goal of the observations. A balance between completeness, the percent of real events actually found, versus the false alarm rate must be made. The program should also be optimized for the object of interest. Searching primarily for supernova and GRBs, the LOOC UP project could modify existing software for its needs, or simply use the operating telescope group's software. It might also need to develop its own highly specific code. This outstanding question must be investigated further. It depends on how wide a field must be surveyed, the desired completeness versus purity tradeoff, how rapidly transients should be identified, and if the software should look for objects other than GRBs and supernova. Since no other GW triggered surveys have been conducted, some trial work should be expected.

Common to all possibilities is the need to eventually test the image processing software. We set out to accomplish this goal by producing a sequence of images containing realistic transient objects. As further motivation, these simulations could also be used as a training dataset for the object detection software. A study carried out by Bailey et al. investigated several advanced methods of transient object classification. They conclude that a boosted decision tree algorithm produces the best results, and yields a much lower false detection rate than the methods employed by most existing software [obj class citation]. This tree takes a wide set of parameters describing possible events as input and sorts through them in a decision tree to determine if the event is real or background. A set of training images must be used to produce the tree. The study used 5,000 real transient events and 5,000 background events to train their tree. Existing freeware [footnote StatPatternRecognition] creates a decision tree that maximally separates the two groups of the training set. Since the performance of the tree is highly dependent on the training set, the training set should be as realistic as possible. It should also simulate exactly the type of events anticipated.

Possible Sources

Before discussing the procedure of the simulation, the type of object we are searching for and therefore simulating must be clarified. The astrophysical source of GWs is predicted to come from stellar core collapses, the coalescence of binary neutron stars, or the coalescence of a neutron star and a black hole.[Saulson]. This translates into looking for supernova and GRBs.

With respect to supernova, LUMIN should look for type II supernova as they are core collapse supernova [guillaume paper]. GRBs are categorized as long GRBs and short hard bursts (SHBs). Long GRBs are also generated from the collapse of a massive star and are possible GW sources [nakar]. SHBs are perhaps of even more interest. Their origin is speculated to be from the merger of neutron stars or a neutron star and a black hole. This would make them the ideal candidate for GWs [nakar]. However, this origin is still debated. Detection of a SHB optical afterglow in response to a GW detection would therefore be a major discovery for the astronomical community as it could conclusively determine their origin.

The key identifying feature of these objects is their light curve. During its cataclysmic collapse, supernova emit 10^49 ergs of electromagnetic radiation. As a result, the supernova appears as a sudden bright spot on the sky which peaks in luminosity then slowly fades away over the span of around one hundred days. See figure 2 for a characteristic light curve. Measurement of the supernova once every few nights to determine its magnitude is sufficient to produce a light curve of the object and identify it as a supernova. Supernova are the most well known of the three targeted events and the easiest to identify.

GRBs are more difficult to find. These events are first discovered by their highly beamed emission of gamma ray radiation. Satellites such as SWIFT are programmed to detect such gamma rays, locate its origin in the sky, and send that information to robotic ground based telescopes such as TAROT. Through this mechanism astronomers have discovered a characteristic afterglow of both long GRBs and SHBs. [tarot paper 2]. This afterglow is what the LOOC UP project hopes to observe and identify as a GRB.

Simulation

The transient object simulations attempt to imitate exactly what a telescope would produce if it were to observe a supernova or GRB. Accordingly, the medium for such simulations is a real background telescope image in the FITS format. These files contain a matrix of values in the unit ADU which correspond linearly to the number of photons detected by each CCD of the telescope. Again, this image is assumed to have been pre-processed with flat fielding, bias subtraction, etc. Put simply, the simulation injects a fake object into this background image scaled to the proper magnitude. In order to simulate the objects' characteristic light curve, a series of images must be produced with injections on each.

The procedure of the simulation is as follows:

1) Read in a sequence of telescope images and pick a good model star on the first image.

2) Follow this model star through each image, storing its locations. All injections will copy its point spread function (PSF) and location displacements.

3) Read in a user provided light curve and adjust it to a new specified distance.

4) Interpolate the light curve according the timestamp on each image and make injections of the corresponding magnitude.

The first two steps aim to greatly increase the realism of the simulation. Any injection requires choosing a PSF. Supernova and GRBs in practice are point-like and thus resemble the PSF of other stars when found in images. A model star is therefore chosen in order to replicate its PSF. An example of a star's PSF is shown in figure 4. By following this model star through each image, the simulation incorporates all noise and complications involved with a changing PSF. Additionally, the injection locations of the transient object are set at a fixed distance in pixels from the model star on each image. This incorporates the fluctuation in the position of stars in the image as well as the deviations that can occur with an individual star.

Choosing a reasonable model star to follow requires careful selection. For example, it must not be part of a binary star system, it must be a star not a galaxy, and it must not fall out of view in any image. This simulation runs the freeware SExtractor on each image to classify all objects in the sky. SExtractor performs aperture photometry on every identifiable object and creates a database of other relevant object properties. From this, the simulation considers each object's full width half maximum, location, PSF template size, star-like categorization, and possible defects such as image saturation as the selection criteria. A set of threshold cuts narrows the star field to a list of reasonable stars. From this list, a star is chosen at random to follow. Consequently, calls to the simulation with the same input parameters can and will yield different simulations of the same light curve.

Following the star also requires use of SExtractor catalogues. The brightest object on each image and its location is identified from the catalogues in order to align each consecutive pair of images. After alignment, a procedure searches the SExtractor output for the object closest to the new anticipated location of the model star. This is then regarded as the same star. In practice, the location of the star rarely exceeds a displacement of 2 pixels from its expected location. As a safety check, the change in magnitude of the object is computed and printed to the screen. A warning message is produced if the change in magnitude exceeds .5 as the code may have failed to track the same object in this case.

Most critical to the simulation is the light curve to be simulated. This project uses optical light curves from real supernova and GRBs found in various scientific papers.[citations]. [footnote – of particular note is the SUSPECT website for supernova light curves]. Nearly all observed light curves come from objects existing well beyond the detection limits of the GW observatories. For GRBs, the upper limit on GW detection is around 15 Mpc, while for supernova this threshold is closer to 10 Kpc. An accurate simulation of events with detectable GWs should not simulate objects much beyond these thresholds. All light curves can therefore be scaled to a new user-input distance according to the following formula:

$$mag_{new} = mag_{old} + 5 * \log\left(\frac{distance_{new}}{distance_{old}}\right)$$

For example, supernova 1999em was observed at a redshift of z = .0024. [source] Assuming a flat universe with the Hubble Constant = 72 km/s/Mpc, Ω m = .27, and equation of state parameter w = -1 as done by Hjorth[cite], this corresponds to a luminosity distance of 10.01 Mpc. The magnitude of its first observation was 13.869. Scaled to a luminosity distance of 10 Kpc, the observed magnitude would be -1.13. This procedure ignores galactic extinction.

With the light curve and model stars determined, the program runs through each image making injections. The correct magnitude for each injection is computed from the light curve and the observation time of the image. Each FITS file contains a standard header which includes its observation time. The time of the first image is scaled to coincide with the first data point of the light curve. All subsequent images rely on interpolation of the light curve. If any image extends beyond the time span of the light curve, an error message is produced.

As mentioned earlier, the injections base their PSF off of a model star. This PSF is deduced by running a recursive algorithm on the star's central location which cuts out all connected pixels above a certain threshold. This threshold is set to 2 sigma deviations above the mean image background brightness. The number of sigma deviations used is left as user-input, however nothing below 1.5 is recommended.

Once a template PSF of values in a 2D matrix is extracted, it is scaled to the correct magnitude according to the following formula:

$$TotalADU = 10^{\frac{cmagr - magnitude}{2.5}}$$

$$A_i = (P_i - bgmean) * \frac{TotalADU}{\sum P_i - bgmean}$$

Which are derived from the photometry equation:

$$magnitude = cmagr - 2.5 * \log\left(\sum P_i - bgmean\right)$$

In the formula, *TotalADU* is the total number of ADU above the mean background value, *bgmean*, necessary to produce an object of the magnitude *magnitude*. *cmagr* is a calibration constant given in the FITS file header, P_i is the value of the i_{th} pixel from the template PSF and A_i is value of the corresponding pixel to be injected. The matrix of values A_i to be injected is then simply added to the existing background on which it is placed. This introduces any random fluctuations in the chosen background region into the object injection.

Once an injection is made, the new star field is automatically written to file and saved. A sequence of simulated images is thus written to disk and ready to be used.

Improvement

The simulation is not in its final form. The code should be considered a work in progress ready to adapt to intelligent ideas for improvement. While the simulation currently makes injections verified by SExtractor to within .05 of the desired magnitude, other aspects can be made more realistic and suited towards large scale applications. This is especially true if a training data set needs to be produced for image processing software.

One current drawback is the manual effort required to find and retrieve a sequence of FITS images of the correct observation schedule desired. This process can be lengthy and downright impossible if several thousand simulations are required. A script should therefore be written to automatically search for image sequences from a database with certain specified properties and download them. Preliminary steps towards this goal have begun.

With respect to tracing a model star improvements can again be made. Tracing the same star through each image is not currently a guaranteed procedure. While it works well with one set of telescope images tested, more tests need to be conducted to assess its true capability. If the method has an unsatisfactory failure rate, more sophisticated image alignment processes should be used. For example the ISIS software package should be investigated for use.

Once again for the purpose of large scale production of simulations, creating the user provided light curves could become a time intensive step. Users should still aspire to use actual light curve data, however a light curve generator may need to be written. Since GRB afterglows typically follow the model of power law decay, $flux \sim t^{-alpha}$, [cite] variations of this tendency could be used to automatically produce large numbers of GRB light curves. Another procedure could generate light curves similar to the databases of supernova light curves.

Further investigations about the light curve accuracy should also be carried out. The light curve information obtained thus far comes from a variety of R, V, r' and other optical filters. The telescope LOOC UP will ultimately use to make its observations will make use of its own filters, and inevitably be different from some of the ways the source light curves were obtained. Whether or not the effects of using different optical filters is significant needs to be investigated. In addition, the light curves also suffer from inaccuracies from placing supernova and GRBs at new distances. While the scaling equations used are mostly correct, they do not take into account galactic extinction. The lights from more distant objects are more likely to be dimmed and distorted by intervening matter than close objects. This effect should also be studied.

Future Work

As the LOOC UP project develops, several important questions will need to be addressed relating to telescope observation and image processing. The most immediate problem is to determine the observation schedule of the telescope. Whether a supernova or GRB is anticipated makes an enormous difference. Based on their light curves, supernova can be observed once every few nights and still be identified. GRBs, however, can occur and fade away on the time scale of minutes and therefore require a much more active observation schedule. If signatures in the GW signal are not strong enough to decide which type of object to look for, one conceivable strategy would be to use a standard schedule to search for GRBs initially and then perform less frequent observations to look for supernova.

Following such a strategy could become costly. The telescopes have their own routines to follow and cannot be overly bogged down by requests for searches from the GW community. High performance image processing software could help alleviate this problem by finding transients in real time. If no interesting transient objects are identified, the telescope could stop observation and only dedicate serious time to observing real possibilities. Real time image processing could also pinpoint the exact location of transients in time to refocus a narrow field telescope on that location. These possibilities require robust software that will take time to develop. Whether the approach is to rapidly identify transients or not also affects the demands on the simulation software. Realistically modeling short scale events poses a different problem from modeling long events since the object changes in magnitude drastically less. As a result random errors in the injections become more significant.

Going even further, the possibility for finding GW associated events that do not look like supernova or GRBs should not be discarded. This mindset would lend itself to using a more routine observation schedule rather than one that aggressively tries to identify objects and adjust accordingly.

Conclusion

Following GW triggers with searches for transient objects in the electromagnetic spectrum opens possibilities for exciting advances in both astronomy and GW physics. A well functioning system could effectively extend the detection range of LIGO and VIRGO observatories. Detection of a SHB in response to a gravity wave could also provide keen insight in the unresolved debate over the origin of these objects. Routine detection, perhaps possible after advanced LIGO in 2015, could become a breakthrough tool for GRB research. Indispensible to this project is automated software to identify possible GW sources. Its performance can have direct influence on LIGOs increased sensitivity. It must therefore be thoroughly tested with simulations made as realistic as possible. While the exact nature of the program remains unknown, the simulation code presented in this paper serves as an adaptive base to test a wide variety of possibilities. The simulations could even become the most essential element of developing image processing code as the training data set.

However, before this prospect can be reached, many decisions related to observations must be overcome. Realistically, a simple transient object identifier should be implanted for the first runs. Only after sufficient experience and data collection can more informed decisions be made.