LIGO SURF 2016 Progress Report 2: The effect of orbital eccentricity in binary black hole simulations on gravitational waveforms

Halston B. Lim
Mentor: Dr. Mark Scheel[1]
[1] *Theoretical Astrophysics 350-17, California Institute of Technology, Pasadena, CA 91125, USA*
(Dated: August 17, 2016)

SUMMARY

In this report, we will review the progress made in this SURF project over the past 3 weeks, from July 7 to July 27, 2016. The first section will focus on the status of the computational simulations, and the second section will focus on the status of the gravitational wave analysis techniques that we will eventually apply to the data.

BACKGROUND

Binary Black Holes and Gravitational Waves

The theory of General Relativity predicts gravitational wave (GW) emission from binary systems consisting of compact objects such as neutron stars and stellar mass black holes (BH). This prediction was recently verified by the Laser Interferometer Gravitational-Wave Observatory (LIGO), which detected GW emission from several binary black hole (BBH) mergers [1]. Once detected, such GW signals can be used to infer source information, such as mass ratios and spins, by a process known as parameter estimation. However, the detection of GWs relies on high quality, accurate theoretical waveforms. Such waveforms can be either computed with analytical approximations such as post-Newtonian theory (PN) or calculated directly by numerically solving the full set of Einstein's equations.

For isolated systems, the orbit of a compact binary system will gradually become circular due to gravitational wave emission. When the GWs reach detection frequency, the eccentricity is expected to be negligible, and therefore only circular waveforms are used to search for GW signals in LIGO data. Since the PN approximation is a perturbative solution which solves Einstein's equations in powers of v/c, when objects are very close together (and $v/c << 1$ is not necessarily true), the PN waveforms are no longer accurate. In these strong field regimes, numerical relativity can provide a solution with an accuracy limited only by available computational resources.

Unlike PN theory, in the full relativistic theory, the initial velocities corresponding to a quasi-circular orbit cannot be computed in closed form. One way to determine these velocities is to start with a reasonable guess, run a simulation long enough to measure the eccentricity, and then compute an updated guess. This is currently implemented in the Spectral Einstein Code (SpEC), a multi-domain pseudo-spectral evolution code originally developed by Lawrence Kidder, Harald Pfeiffer, and Mark Scheel, that, given initial spins and a mass ratio, calculates inspiral orbits, merger, and ringdown of compact binaries [2].

The GWs emitted from eccentric binaries are expected to be different from non-eccentric (circular) binaries - namely, the frequency content of eccentric waveforms is more complicated than that of non-eccentric ones [3]. Also, the peak emitted power will be greater due to a greater orbital apastron, leading to more dynamical motion. Therefore, if the BBHs LIGO detects are truly circular, then numerical relativity waveforms should also derive from *effectively* circular orbits. Since the eccentricity reduction scheme used by SpEC results in orbits that have $e \approx 10^{-4}$, it is important to determine how small the eccentricity of a numerical relativity simulation must be in order to justify treating it as zero.

RESEARCH GOALS

The primary objectives of this SURF project are:

- First, to determine the effect of eccentricity on the GW emission from BBHs and to compare the properties of the waveforms using some distinguishability criteria.
- Second, to investigate the experimental implications of using eccentric waveforms on detection and parameter estimation. We will aim to determine the smallest orbital eccentricity required in order to be experimentally indistinguishable from zero eccentricity.
- Third, to learn about the computational techniques used in numerical relativity. More specifically, to gain experience with the SpEC evolution code in application to BBH inspirals.

SIMULATION PROGRESS

To generate a set of otherwise-identical SpEC waveforms with different orbital eccentricity, we use SpEC's automatic eccentricity-reduction scheme. This scheme, however, must be modified so that each waveform at each different eccentricity proceeds all the way to merger and ringdown. We proceed in two stages:

1. **Stage 1: Initial Job Submission** First, the SpEC simulation is run with the iterative eccentricity reduction scheme. For each iteration, SpEC first evolves the system a few orbits, then measures the eccentricity and adjusts the initial separation of the BHs, the initial time derivative of the separation, and the orbital frequency such that the next iteration has a smaller resulting eccentricity. This continues until the eccentricity is less than some target (typically 10^{-5}) or fails to decrease with successive iterations. There are usually 3-5 iterations in this process, and are referred to has "Ecc0", "Ecc1","Ecc$n$" when mentioning the zeroth, first, nth iteration.
2. **Stage 2: High Eccentricity Re-submission** Second, after the SpEC simulation completes the eccentricity reduction scheme, there will be several incomplete orbits of successively smaller eccentricity. For example, run 7.6 completed stage 1 within 3 iterations - "Ecc0", "Ecc1", "Ecc2" - each with successively smaller eccentricity. Each iteration, which contains only a small number of orbits, must be restarted but without the eccentricity termination condition, so that it runs all the way through merger and ringdown. This is achieved by modifying some input files using a script that the author wrote. Each iteration will also be simulated at multiple resolutions, which we call Lev1, Lev2, and Lev3, in order of increasing resolution. Thus, the total number of SpEC jobs in stage 2 for run 7.6 is 9 since there are three resolutions for each of the three iterations.

The status of the stage 1 jobs is summarized in table I.

Name	Status	Iterations of Ecc. Red.
7.1	Finished	Ecc0, Ecc1, Ecc2, Ecc3, Ecc4
7.2	Running	Ecc0, Ecc1, Ecc2, Ecc3, Ecc4, Ecc5
7.3	Finished	Ecc0, Ecc1, Ecc2, Ecc3, Ecc4
7.4	Finished	Ecc0, Ecc1, Ecc2, Ecc3, Ecc4
7.5	Running	Ecc0, Ecc1, Ecc2, Ecc3
7.6	Finished	Ecc0, Ecc1, Ecc2

TABLE I. Summary of stage 1 SpEC runs. The spins and mass ratios for each run are mentioned in table 2 of progress report 1, which is reproduced as table III in the appendix. For simulations in progress, the number of eccentricity reduction iterations may still increase, if the eccentricity reduction scheme undergoes another iteration.

The status of stage 2 jobs is summarized in table II. Only runs which have finished stage 1, and have a known number of eccentricity reduction iterations, are listed. Most stage 2 jobs, even though prepared for submission on a computer cluster are left unsubmitted because there is a new, faster Caltech computer cluster, Wheeler, that we anticipate utilizing in the very near future.

GRAVITATIONAL WAVE ANALYSIS PROGRESS

Week 1: July 7 - July 12

The majority of week 1 was dedicated towards identifying and fixing a bug in *GWFrames*[4]. The module *GWFrames* contains many useful operations on time series waveforms including noise weighted inner products, derivatives, and

Name	Stage 1 Name	Status (Lev1, Lev2, Lev3)	Evolution time (M) (Lev1, Lev2, Lev3)
11.1	7.1 Ecc0	(Idle, Idle, Idle)	(2447.2, 541.78, 1992.5)
11.2	7.1 Ecc1	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.3	7.1 Ecc2	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.4	7.1 Ecc3	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.5	7.1 Ecc4	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.6	7.6 Ecc0	(Running, Idle, Running)	(1857.6, 1667.6, 2656.5)
11.7	7.6 Ecc1	(Running, Idle, Idle)	(1811.8, 3351.2, 2993.4)
11.8	7.6 Ecc2	(Running, Idle, Idle)	(2067.3, 2367.7, 2907.8)
11.9	7.3 Ecc0	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.10	7.3 Ecc1	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.11	7.3 Ecc2	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.12	7.3 Ecc3	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.13	7.3 Ecc4	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.14	7.4 Ecc0	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.15	7.4 Ecc1	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.16	7.4 Ecc2	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.17	7.4 Ecc3	(Unsub, Unsub, Unsub)	(0., 0., 0.)
11.18	7.4 Ecc4	(Unsub, Unsub, Unsub)	(0., 0., 0.)

TABLE II. Summary of stage 2 SpEC runs. The status column and the evolution time column contain the status and current evolution time of the three resolutions to date. In the status column: *Idle* refers to a job in the queue and waiting to run, *Unsub* refers to a job that has been prepared for submission (e.g. eccentricity termination condition removed) but has not been submitted yet.

fourier transforms. For this SURF project, we utilize a match function, which, given two waveforms expressed in the frequency domain, outputs the maximum inner product, optimized over extrinsic parameters (relative time and phase of coalescence) and normalized. Further details are in progress report 1. The match function is defined in *WaveformsAtAPointFT.hpp* in the *GWFrames* module and has the following function call:

void Match(const WaveformAtAPointFT& B, const std::vector⟨ double⟩& InversePSD, double& timeOffset, double& phaseOffset, double& match) const;.

Here, the match function aligns two gravitational waveforms, the first being the WaveformAtAPointFT object (from which Match is called) and the second being the WaveformAtAPointFT object passed into the match function, B. Several problems quickly surfaced when trying to read out the time offset and phase offset from the match function, including undocumented sign conventions, and a bug in calculating the time offset that the author tracked down and fixed. However, instead of describing all the difficulties along the way, we will just describe what we learned and how to properly use the match function.

Suppose that there are two waveforms - $A = A(t)$ and $B = B(t)$. Then the match function will return the time shift in seconds, and the phase shift in radians, with sign convention depending on from which *WaveformAtAPointFT* object the match function is called. Precisely, if the match function is called as: A.Match(B, InversePSD, $\Delta\tau$, $\Delta\phi$), then $\Delta\tau$ and $\Delta\phi$ are such that

- $A(t - \Delta\tau)e^{i\Delta\phi} = B(t)$, and
- $\widetilde{A}(f)e^{-i(2\pi f\Delta\tau - \Delta\phi)} = \widetilde{B}(f)$.

Week 2: July 13 - July 19

Week 2 was devoted towards preparing stage 1 jobs for stage 2 submission, including writing a script which automatically copies files from a finished stage 1 project folder to a designated stage 2 folder, while making necessary modifications to the files. To be most efficient, stage 2 jobs should be submitted as soon as the stage 1 simulation is finished (as opposed to waiting for all the stage 1 jobs to finish). Thus, the ordering of the stage 2 jobs as listed in table 2 (11.1, 11.2, ...) was determined by the order in which the stage 1 jobs finished.

Here we describe technical details of how to use the Binary Factory Infrastructure (BFI) to submit the stage 2 jobs successively without an eccentricity termination condition. The user should copy all the stage 1 files to a new folder and make the following changes:

- Ensure a parallel folder structure for stage 1 and stage 2 jobs. That means that even though each stage 2 job has only one iteration (in stage 1, each job had multiple eccentricity reduction iterations), there still needs to be a folder called *Ecc0* in the stage 2 job path
- The target eccentricity in *Evolution.input* must be made big (a value of 1 works)
- *RunsDatabaseInfo.input* must point to the most recent Lev run folder in stage 2 project, instead of the stage 1 project
- Within the stage 2 project folder, the jobname in *MakeSubmit.input* must be different from the old name used for stage 1
- Modify *DoMultipleRuns.input* so that the variable $EccRedRun = 0.
- Change *TerminationReason.txt* from EccentricityReduction to WallClock
- Run *MakeSubmit.py update* to reflect any changes made
- Run *DoMultipleRuns -n* to trick SpEC into making folders for Lev1 and Lev2 jobs, even though only Lev3 folders may exist.

Week 3: July 20 - July 26

During week 3 we investigated an artifact in the GW output, which appeared to be a bug. Once a SpEC simulation is complete, the GW signal at infinity is extrapolated from the GW signal at successive finite radii. However, since the GW signal can be decomposed into a linear combination of spherical harmonic modes, labeled as (l, m), the time domain waveform can be extracted in two ways:

1. The total GW signal, or any subset of harmonics, emitted in a particular direction, (θ, ϕ) - the inertial coordinates of the BBH system
2. A particular mode's contribution to the total GW signal, over all directions

We expect that the waveform should change continuously over all propagation directions, but the initial version of *GWFrames* produced discontinuous changes in the Fourier transform of the waveform at certain (θ, ϕ) values. The source of the discontinuity was in the windowing function used to find the Fourier transform of a time domain waveform, inside *WaveformsAtAPointFT::WaveformAtAPointFT*. This was another bug in *GWFrames* that was found by the author. Currently, changes to *GWFrames* are being implemented, so that the windowing function no longer implicitly depends on the waveform direction.

As a next step for this project, we will determine how the the length of the (time domain) gravitational waveform impacts the GW mismatch between otherwise identical BBHs with differerent eccentricities. One way to approach this problem is to find the dependence of the GW mismatch on a low frequency cut off, so that the beginning of the waveforms are excluded.

[1] B. P. Abbott et al. Observation of gravitational waves from a binary black hole merger. *Phys. Rev. Lett.*, 116:061102, Feb 2016.

[2] Luisa T. Buchman, Harald P. Pfeiffer, Mark A. Scheel, and Béla Szilágyi. Simulations of unequal mass binary black holes with spectral methods. *Phys. Rev. D*, 86:084033, 2012.

[3] P. C. Peters and J. Mathews. Gravitational radiation from point masses in a Keplerian orbit. *Phys. Rev.*, 131(1):435–440, Jul 1963.

[4] Michael Boyle. Angular velocity of gravitational radiation from precessing binaries and the corotating frame. *Phys. Rev. D*, 87(10):104006, May 2013.

Name	q	$\lvert\chi^A\rvert$	χ^A_θ	χ^A_ϕ	$\lvert\chi^B\rvert$	χ^B_θ	χ^B_ϕ
7.1	3.0	0.7	0	0	0.6	0	0
7.2	3.0	0.7	3.14159	0	0.6	3.14159	0
7.3	3.0	0.7	1.0	0.5	0.6	0.5	1.0
7.4	3.0	0.7	1.0	0.5	0.6	2.0	4.0
7.5	2.0	0.0	0.0	0.0	0.0	0.0	0.0
7.6	1.0	0.0	0.0	0.0	0.0	0.0	0.0

TABLE III. Summary of stage 1 SpEC runs. Here, q is the mass ratio and χ^X_μ is the μ-component of the initial spin vector for black hole X. Given the magnitude of the spin vector, $|\chi^X|$, and two components $\chi^X_\theta, \chi^X_\phi$, the third can be determined.