LIGO-T1500606-v5

# Numerical Relativity Injection Infrastructure

Patricia Schmidt,[1,2] Ian W. Harry,[3] and Harald P. Pfeiffer[3,4]

[1]LIGO Laboratory, California Institute of Technology, MS 100-36, Pasadena, California 91125, USA
[2]Theoretical Astrophysics, Walter Burke Institute for Theoretical Physics,
California Institute of Technology, Pasadena, California 91125, USA
[3]Max Planck Institute for Gravitational Physics (Albert-Einstein-Institute),
Am Mühlenberg 1, Potsdam-Golm, 14476, Germany
[4]Canadian Institute for Theoretical Astrophysics, 60 St. George
Street, University of Toronto, Toronto, ON M5S 3H8, Canada

This document describes the new standardised NR injection infrastructure in the LIGO Algorithms Library (LAL), which henceforth allows for the usage of Numerical Relativity (NR) waveforms as a discrete waveform approximant in LAL. With this new interface, NR waveforms provided in the appropriate format can directly be used as mock GW signals ("injections") for LIGO data analyses, which include parameter estimation, searches, hardware injections etc. The interface handles higher-order modes making them directly accessible to LIGO analyses for the very first time. The NR injection infrastructure also takes care of waveform frame conventions and transforms the NR data into the correct LAL frame.

## I. INTRODUCTION

LIGO reported the first detection of gravitational waves (GW) from a merging binary black hole system [1]. Such coalescing compact binaries are prime sources of GWs for ground-based interferometric GW detectors and play a critical role in LIGO data analysis. Whilst for low-mass systems only the early part of the binary evolution, the inspiral, is accessible to LIGO, for high-mass systems also the later stages, in particular the merger and ringdown of the final black, are visible in LIGO's sensitivity band.

During the early part of the binary coalescence, the emitted gravitational waveforms are accurately described by analytic post-Newtonian expansions of the Einstein field equations [2]. To obtain the waveforms for the final stages, the full non-linear solutions of the field equations are required, which are provided by Numerical Relativity (NR) (see for example [3] for an overview).

One of the most important impact of numerical relativity on LIGO data analysis is the construction of waveform models for the complete inspiral-merger-ringdown (see [4] for a review). Such waveform models [5, 6] underpin LIGO's GW searches, parameter estimation and tests of general relativity ([1] and references therein). In the analysis of GW150914, numerical simulations were also used to determine the remnant black hole's mass and spin [7], and to investigate systematic biases due to waveform modeling errors.

In the Advanced detector era, it is very advantageous to be able to directly use NR waveforms in gravitational-wave searches and parameter estimation, to test General Relativity and to assess the systematics of analytic waveforms models within a uniform framework. Such use of NR waveforms should be easy and convenient. This is now possible with the new *Numerical Relativity Injection Infrastructure* described in this document. Once the NR data is provided in a specific format (as described below), this infrastructure allows for the treatment of NR waveforms as a "discrete" waveform approximant, which can seamlessly be called from within the LIGO Algorithm Library (LAL).

In previous efforts, binary-black-hole (BBH) hybrid waveforms constructed by combining a PN inspiral with an NR merger-ringdown waveform, were used in LIGO data analysis and parameter estimation in the NINJA and NINJA-2 projects [8, 9]. However, the previously employed NR modules in LAL required the NR waveforms to be resampled to a uniform time-spacing. In the NINJA framework, the resampling was performed *before* inserting the total mass scale. For the waveforms to be useable at high total mass, the time-spacing in the NR data had to be very small, resulting in very large storage requirements, even if only the dominant harmonics $(\ell, m) = (2, \pm 2)$ were supplied.

The new infrastructure described here improves on the earlier approaches in several significant ways. First, data is stored in a highly efficient compressed format; even including a large number of sub-dominant modes (as is now encouraged), storage requirements are lower than just for the dominant modes in the preceding storage format. Second, the compressed NR data are interpolated with 1D spline interpolation *after* the mass scale is inserted. This avoids high-memory operations and further reduces storage requirements and I/O times. Third, the new infrastructure handles projection of the NR data onto arbitrary source-location and detector orientation using the same conventions in LAL as other waveform families. The new injection infrastructure is fully implemented in LAL and is intended to supersede previously used NR modules.

The remainder of this technical document is organised as follows: In Sec. II we provide a brief summary of the NR data format and metadata required as input. In Sec. III we describe the basics of the waveform generation code and give explicit examples of how the NR waveforms are evaluated in `lalsimulation`. Sec. IV details the frame transformations between the NR simulation frame and the LAL wave frame. We highlight caveats and desired future improvements in Sec. V.

## II. WAVEFORM FORMAT

In Numerical Relativity one solves for the complete space-time of the binary system. For data analysis purposes however, one requires the gravitational-wave strain $h(t)$ far from the source. The relevant numerical quantity is the metric perturbation $h_{ij}^{\mathrm{TT}}$ as computed in the transverse-traceless gauge (TT).

There are different way of computing the metric perturbation from a numerical evolution. The most common methods include the use of the complex Weyl scalar $\Psi_4$, which is related to the metric perturbation via two time derivatives, or the Regge-Wheeler-Zerilli formalism, which computes the metric perturbation in the wave-zone as a perturbation of the Schwarzschild spacetime.

In the TT gauge, the metric perturbation has two independent real polarisations, $h_+$ and $h_\times$, which can be written as a complex strain by

$$h = h_+ - i h_\times \in \mathbb{C}, \tag{1}$$

where $h_+, h_\times \in \mathbb{R}$.

Let $(t, x, y, z)$ be a Cartesian coordinate system in the wave-zone, which is related to the polar coordinates $(r, \theta, \phi)$ by the standard means. In this coordinate system, the metric perturbation is commonly decomposed into *modes* in a basis of spin-weighted spherical harmonics, ${}^{-2}Y_{\ell m}$, of spin weight $s = -2$, where the GW propagation direction is the radial unit vector r. For any point $(\theta, \phi)$ on the unit sphere, the GW strain can then be shown to take the following form:

$$h^{\mathrm{NR}}(t; \theta, \phi) = h_+^{\mathrm{NR}} - i h_\times^{\mathrm{NR}} = \sum_{\ell=2}^{\infty} \sum_{m=-\ell}^{\ell} H_{\ell m}(t) {}^{-2}Y_{\ell m}(\theta, \phi). \tag{2}$$

As for any wave, we can also write each NR gravitational-wave mode $H_{\ell m}$ as an amplitude $A_{\ell m}$ and a phase $\Phi_{\ell m}$,

$$H_{\ell m}(t) = A_{\ell m}(t) e^{i \Phi_{\ell m}(t)}. \tag{3}$$

The amplitude of each mode is its complex norm, the phase is the unwrapped argument of the complex time series $h_{\ell m}$. For a binary that is orbiting counter-clockwise in the xy-plane (i.e. orbital angular frequency vector is *parallel* to the z-axis), $\Phi_{2,2}(t)$ is a monotonically **decreasing** function.

Following the LAL waveform convention, the time-coordinate $t$ in the waveform modes should be chosen such that the peak of the waveform occurs at $t_{\text{peak}} = 0$. We recommend to define "peak" of the waveform by

$$h_{\text{peak}} = \max\left(\sum_{\ell,m} A^2_{\ell,m}\right). \tag{4}$$

### A. Spline Compression

Gravitational waveforms for LIGO data analysis purposes require uniform sampling in time for a given sampling frequency $f_s$. NR datasets however, are commonly not uniformly sampled and if they are, the sampling interval $dt$ may not necessarily correspond to the one required by data analysis tools. It is therefore unavoidable to interpolate the NR data to the desired sampling rate. Whilst the NR data could simply be interpolated as they are, we choose to reduce the data by performing 1D spline compression on the NR data [11]. This is a particular advantage for long simulations or hybrid PN-NR data, but also significantly reduces the storage and I/O for pure NR data.

The 1D spline compression is performed separately for each **mode amplitude** $A_{\ell m}$ and **phase** $\Phi_{\ell m}$, which are already time-shifted such that $t = 0$ corresponds to the peak of the waveform. The routine employed to compress and interpolate the data is the *reduced-order spline interpolation* presented in [11]. It uses a greedy algorithm, which selects the near-optimal points to construct a univariate spline interpolant with a specified global accuracy. By default, the interpolants are constructed using fifth degree polynomials and a tolerance of $10^{-6}$, i.e., if the spline is evaluated at the discrete times $t_i$ of the NR data, which were not selected by the greedy algorithm, the original NR values are recovered with an error equal to or smaller than the specified tolerance. In addition, a median error can be associated with any predicted value, i.e., a value not contained in the original NR data. For a detailed description of this method and the accuracy of the obtained interpolants, we refer the reader to [11]. The spline compression is conveniently performed using the publicly available Python package `romSpline` by Chad R. Galley [12].

The spline interpolants for each amplitude and phase are obtained via `romSpline.ReducedOrderSpline()` as follows:

```
import romSpline
spline = romSpline.ReducedOrderSpline($t$, $A_{\ell,m}(t)/\phi_{\ell,m}(t)$, verbose=False)
spline.write('filename.h5')
```

The output of `spline.write` contains all information needed for subsequent interpolation of the respective input time-series. Note: a newer version of romSpline now also allows to pass a group descriptor, so that the spline-data can be written directly into the appropriate group of the final output file, without the need for an intermediate step.

Once the spline interpolants have been obtained, they have to be stored as individual groups in a *single HDF5 file* with the following mandatory group naming conventions:

- Amplitude group: **amp_l#1_m#2**
- Phase group: **phase_l#1_m#2**,

where (#1, #2) are placeholders for $(\ell, m)$, e.g., for $(\ell = 2, m = -2)$ the naming convention is phase_l2_m-2 and amp_l2_m-2.

Pure NR data without a hybridised PN inspiral need to have the initial *junk radiation* removed before the spline interpolants are constructed but no tapering should be applied at the beginning or end of the waveform data. In addition to the amplitude and phase groups, we recommend to also include the NR times, i.e., the 1D data array containing the original time-steps of the NR data, as a group in the HDF5 file.

### B. Metadata

The metadata format is adapted from the original NINJA-2 metadata format [10]. The metadata are described in the following list, and are stored as *attributes* of the HDF5 file. Often, more extensive metadata is available for a NR simulation. We recommend that such additional metadata is included as a free-format text data-set the HDF5 file.

In the current implementation, those required metadata which are time-dependent (see list below) have to contain the values associated with the first entry in the stored time series. For pure NR data, since the junk radiation has to be removed, these are *not* the initial data of the simulation. We refer to this very first data point stored in the HDF5 file as the *beginning* of the waveform data. Source frame dependent metadata should be given in the NR simulation frame.

The required metadata attributes are the following:

- ['NR-group']: string to identify the NR group/code
- ['type']: keyword description of the simulation. Allowed values are: aligned-spins, non-spinning, precessing.
- ['name']: alternative simulation name or internal simulation identifier
- ['object1']: keyword description to identify the object type. Allowed values are: BH, NS
- ['object2']: keyword description to identify the object type. Allowed values are: BH, NS
- ['mass1']: mass of the heavier object; if both objects are 'BH', the unit of mass is arbitrary. If at least one object is a NS, then the unit is solar masses $M_\odot$.
- ['mass2']: mass of the lighter object; if both objects are 'BH', the unit of mass is arbitrary. If at least one object is a NS, then the unit is solar masses $M_\odot$.
- ['eta']: the symmetric mass ratio of the simulation
- ['spin1x']: x-component of the dimensionless spin of the heavier object
- ['spin1y']: y-component of the dimensionless spin of the heavier object
- ['spin1z']: z-component of the dimensionless spin of the heavier object
- ['spin2x']: x-component of the dimensionless spin of the lighter object
- ['spin2y']: y-component of the dimensionless spin of the lighter object
- ['spin2z']: z-component of the dimensionless spin of the lighter object
- ['LNhatx']: x-component of the orbital angular momentum unit vector

- ['LNhaty']: y-component of the orbital angular momentum unit vector
- ['LNhatz']: z-component of the orbital angular momentum unit vector
- ['nhatx']: x-component of the orbital separation unit vector
- ['nhaty']: y-component of the orbital separation unit vector
- ['nhatz']: z-component of the orbital separation unit vector
- ['f_lower_at_1MSUN']: frequency of the $(2,2)$-mode in Hz at the beginning of the waveform scaled to 1 solar mass

To reduce potential ambiguities, we further recommend adding the following additional metadata:

- ['eccentricity']: eccentricity parameter of the simulation
- ['mean_anomaly']: mean anomaly at the beginning of the waveform (cf. Eq. 25).
- ['PN_approximant']: string identifier for the inspiral if hybrid data are stored

As indicated, vectors are represented by their Cartesian components. These components must be specified in a coordinate system (referred to as "NR frame") consistent with the one used in the decomposition in spherical harmonics (e.g. the polar axis of the spherical harmonics must point along the +z-axis). No requirements are placed on the orientation of the NR frame. For instance, `nhat` and `LNhat` need not point along specific coordinate axes. Note also that the spin-fields in the metadata `spin{1,2}{x,y,z}` are given in the NR frame, and *not* in the LAL frame. The rotation of the GW modes into the LAL frame is taken care of by the NR injection infrastructure, following the conventions described in Sec. IV.

## III. NR WAVEFORM EVALUATION IN LAL

Once the HDF5 file has been provided, the NR waveforms can be evaluated through the standard waveform interfaces `ChooseTDWaveform()` in LAL and `get_td_waveform()` in pyCBC [13]. The approximant name is "NR_hdf5". The spline data are read from file and evaluated for the desired extrinsic parameters. Since some intrinsic parameters of an NR simulations are fixed (e.g. mass-ratio and dimensionless spins), internal checks on the mass ratio and the spin components are performed to guarantee the consistency between the values passed in the waveform generation call and metadata values.
Note: The waveform generator requires the input spin values to be defined as given by Eqn. (47). In general, these are different to the values of the spin metadata and need to be computed using the metadata for the spins, the orbital angular momentum and the orbital separation.
For a given starting frequency and total mass, a time array is allocated based on an estimate of the waveform length. We use the LAL function `SimIMRSEOBNRv2ChirpTimeSingleSpin()` to estimate the waveform length with an additional leverage of 10%. If the NR waveforms are not long enough for a given total mass and starting frequency, the generation is aborted and error is generated. From the estimated length and the desired sampling rate, the discrete time series for the spline evaluation is determined.

To construct the NR GW polarisations $h_+^{\mathrm{NR}}$ and $h_\times^{\mathrm{NR}}$, the splines for each amplitude and phase are evaluated at the points of the discrete time times, weighted by the ${}^{-2}Y_{lm}$, and summed up accordingly:

$$\mathrm{Re}(H_{\ell m}) = A_{\ell m}\cos(\Phi_{\ell m}), \tag{5}$$

$$\mathrm{Im}(H_{\ell m}) = A_{\ell m}\sin(\Phi_{\ell m}), \tag{6}$$

$$h_+^{\mathrm{NR}} = \sum_{\ell,m} \mathrm{Re}(H_{\ell m})\mathrm{Re}({}^{-2}Y_{\ell m}) - \mathrm{Im}(H_{\ell m})\mathrm{Im}({}^{-2}Y_{\ell m}), \tag{7}$$

$$h_\times^{\mathrm{NR}} = \sum_{\ell,m} -\mathrm{Re}(H_{\ell m})\mathrm{Im}({}^{-2}Y_{\ell m}) - \mathrm{Im}(H_{\ell m})\mathrm{Re}({}^{-2}Y_{\ell m}). \tag{8}$$

Note that all waveform modes $(\ell, m)$ present in the HDF5 file are used to compute the two polarisations. The algorithm will incorporate any mode between $\ell = 2$ and $\ell = 8$ if present in the HDF5 file but at least the $(\ell = 2, m = 2)$ has to be present in the NR data file.

The compressed NR data files do not store the splines themselves, but the knots, errors, the polynomial degree etc. In the pyCBC implementation, the splines are constructed from the HDF5 file via the python function `UnivariateSpline`, in the LAL implementation regular GSL interpolation is performed. Fig. 1 shows an example comparison between NR waveforms obtained using the pyCBC and the LAL waveform generators. The mismatch between the waveforms is less than $10^{-7}$. This is consistent with the level of disagreement expected due to the different numerical interpolation routines.
The source code can be found in `lalsuite/lalsimulation/src/LALSimIMRNRWaveforms.c`.

### A. Examples

There are a variety of different ways to evaluate NR waveforms using LIGO data analysis software. Here, we give two explicit examples for function calls in Python: The first example uses the uses the SWIG-wrapped version of lalsimulation, the send one the pyCBC implementation.

1. Using the implementation in lalsimulation through SWIG:

```
import lalsimulation as lalsim
flags = lalsim.SimInspiralCreateWaveformFlags()
lalsim.SimInspiralSetNumrelData(flags,'/PATH/TO/HDF5')
hp, hc = lalsim.SimInspiralChooseTDWaveform(phiRef,
              deltaT, m1 * MSUN_SI, m2 * MSUN_SI,
              s1x, s1y, s1z,
              s2x, s2y, s2z,
              fStart, fRef,
              distance, inclination,
              $\lambda_1 = 0$, $\lambda_2 = 0$,
              flags, nonGRparams=None,
              amplitudeO=-1, phaseO=-1,
              approximant=lalsim.NR_hdf5),
```

where $\lambda_1$ denote the tidal deformation parameters, which vanish in the case of black holes.

2. Using the pyCBC waveform interface `get_td_waveform()`:

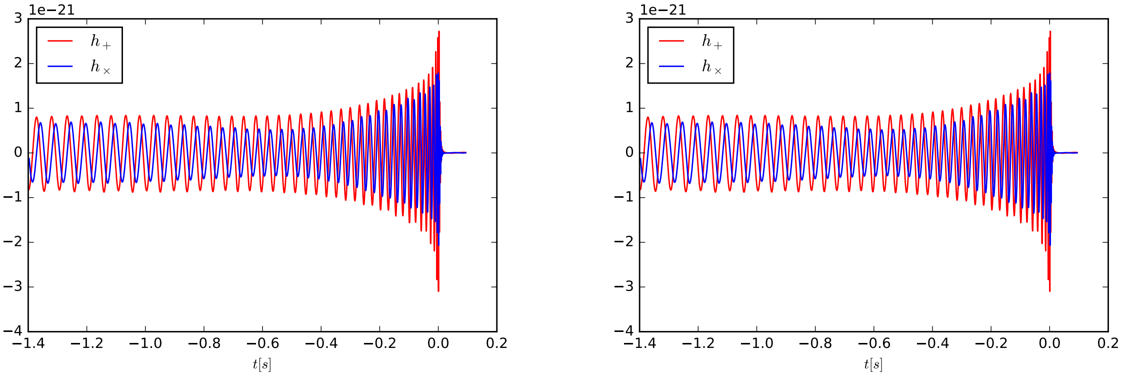

FIG. 1: The waveform polarizations $h_+$(red) and $h_\times$(blue) of the publicly available SXS waveform SXS:BBH:0006 generated via the waveform interface in lalsimulation (left panel) and in pyCBC (right panel).

```
from pycbc.waveform import get_td_waveform
hp, hc = get_td_waveform(approximant=’NR_hdf5’,
                         numrel_data=’/PATH/TO/HDF5’,
                         mass1, mass2,
                         spin1x, spin1y, spin1z,
                         spin2x, spin2y, spin2z,
                         delta_t, f_lower, f_ref,
                         inclination, distance,
                         coa_phase)
```

Fig. 1 shows the two waveform polarizations $h_+$ and $h_\times$ for the precessing binary black hole hole simulated in case SXS:BBH:0006 from the publicly available SXS catalogue [14] for a total mass of 50 $M_\odot$ and an inclination of $\pi/3$. The dimensionless spins for this simulation in the LAL frame are $\vec{\chi}_1 = (-0.05, -0.27, -0.16)$ and $\vec{\chi}_2 = (-0.02, -0.11, -0.10)$ and the component masses are $m_1 = 28.68$ and $m_2 = 21.32$. The waveform is generated from its beginning, corresponding to the starting frequency of fStart=18.76Hz. Further parameters are: distance=100Mpc, deltaT=1.0/16384, phiRef=0 and fRef=fStart.

## IV. LAL COORDINATE FRAMES FOR PRECESSING BINARIES AND NR INJECTIONS

### 1. *Executive summary: changes of conventions*

In recent discussions, it became apparent that current LAL conventions to specify precessing binary compact objects are not ideal: (i) The phase-angle `phiRef` couples specification of the line-of-sight to Earth with specification of spin components `S1x, S1y, S1z, S2x, S2y, S2z`. To compute waveforms for the *identical* CBC system viewed from different directions, one may have to specifiy different values for the in-plane spin-components `S1x, S1y, S2x, S2y`. (ii) Several existing waveform models do not conform to the current LAL convention, already implementing the improvements suggested in this note.

Concretely, it is suggested to specify spin-components in a geometric way based on the angular momentum $\hat{L}$ and line connecting body 2 to body 1, $\hat{n}$:

$$\texttt{S1x} \overset{\text{ref}}{=} \vec{\chi}_1 \cdot \hat{n}, \tag{9}$$

$$\texttt{S1y} \overset{\text{ref}}{=} \vec{\chi}_1 \cdot (\hat{L} \times \hat{n}), \tag{10}$$

$$\texttt{S1z} \overset{\text{ref}}{=} \vec{\chi}_1 \cdot \hat{L}, \tag{11}$$

(and similarly for the second body). The symbol $\overset{\text{ref}}{=}$ indicates that equality only holds at the reference time, as spins generically precess during an inspiral.

Furthermore, it is suggested to specify orientation of eccentric orbits through the angles

$$\texttt{phiRef} = \Phi \overset{\text{ref}}{=} \angle(\hat{n}, \text{line-of-ascending-node}), \tag{12}$$

$$\texttt{mean\_anomaly} = \delta. \tag{13}$$

### 2. *Motivation & benefits of new conventions*

Quite generally waveform modeling requires at least two coordinate systems, a "source-frame" in which it is convenient to specify properties of the source of gravitational waves, and a "wave-frame" which is adopted to wave-propagation to detectors on Earth. Furthermore, numerical relativity data is specified in whatever coordinates are employed during the numerical evolution, generally resulting in a third coordinate system.

This section defines coordinate frames for use in LAL. Specifically, we achieve:

1. Identification of a set of intrinsic parameters that fully describe the dynamics of a binary on an eccentric orbit, defined solely in terms of the source-frame (i.e. independent of the wave-frame).

2. Identification of three angles that describe the transformation between source- and wave-frame, which are independent of the intrinsic parameters.

3. Identification of parameters describing the orbital phase and periapsis location, which have a convenient circular-orbit limit: As the eccentricity tends to zero, one of the two phase-parameters reduces to the standard orbital phase for circular orbits, whereas the other becomes irrelevant.

4. Identities that relate the basis-vectors in the source-frame to the wave-frame (and vice versa). These identities are written in vectorial form and are valid in any coordinate system.

The orthogonal decomposition into intrinsic and extrinsic parameters (points 1 and 2) allows to change the direction at which a binary is viewed (i.e. the wave-frame), without having to adjust the parameters that determine the intrinsic dynamics. Point 3 prepares the ground for easy extension to eccentric waveforms. Point 4 is of particular relevance when translating NR data into LAL-conventions: Evaluating the vector identities in the NR-coordinate system, yields immediately the relation between NR coordinates and wave-frame.

### 3. *NR waveform injections*

NR data are assumed to be supplied in the data-format defined in Sec. II and III, and so it needs to be transformed into the LAL wave-frame. For a given reference time, an entire NR data-set could in principle be transformed into the LAL frame by suitable transformations of each $H_{\ell m}$-mode. Applying such a rotation on the NR-data as a pre-processing step before using it for injections suffers from two disadvantages:

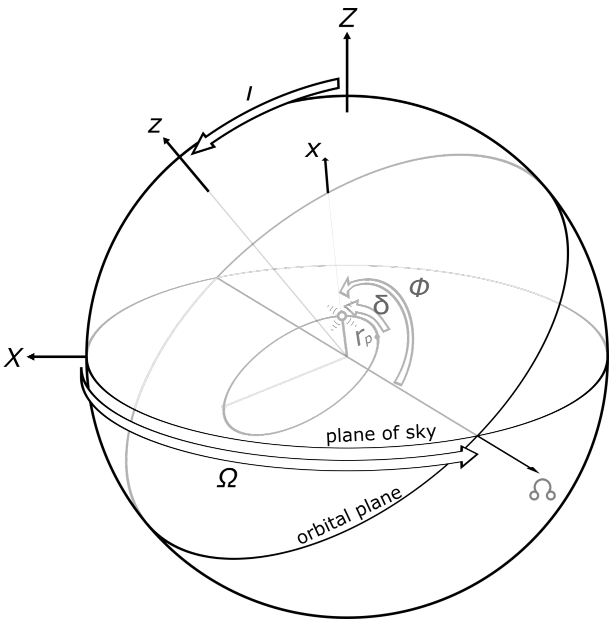

FIG. 2: Coordinate frames. (x,y,z) denotes the source-frame and (X,Y,Z) the wave-frame. For the recommended default $\Omega = \pi/2$, the line of ascending nodes agrees with $\hat{Y}$ i.e. the rotation by the inclination angle $\iota$ is about the $\hat{Y}$-axis.

First, the person who prepares the NR-data for LAL-use must perform the rotation, and must do so correctly. Since NR data is prepared separately by several NR groups, this opens the possibility of introducing errors in this step. Second, for precessing systems, the orbital angular momentum $\hat{L}$ precesses. Therefore, pre-rotating the NR-data locks in the reference point, and one would need to generate different pre-rotated NR-data for a different reference point. Since NR data is in geometric units, the NR-data would have to be separately rotated whenever the total mass of an injection changes.

To avoid both disadvantages, it is proposed to leave the NR-data in its original frame (see Sec. II). Instead, the transformations betweeen the LAL-convention and the NR-data are applied *at use* during the call to `get_td_waveform` or similar (cf. Sec. III).

This section develops the necessary transformations to implement this technique. No assumptions are made on the NR-coordinate system, in order to allow for precession when NR-orbital angular momentum is typically *not* along the z-axis of the NR-coordinate system. The derived formulae also allow to choose an arbitrary reference point for the NR waveform, as long as orbital angular momentum and vector connecting the two bodies are known at that point.

### A. Frames

The frames defined here differ somewhat from the preceding LAL conventions, as spelled out in `http://software.ligo.org/docs/lalsuite/lalsimulation/group__lalsimulation__inspiral.html`. The differences are needed to achieve the separation between intrinsic and extrinsic parameters. In the old conventions, the orbital phase $\phi$ was also used as part of the rotation parameters that define the rotation between source- and wave-frame. Therefore, to "look" at the same binary from different angles used to require a suitable change in the spin-components tangential to the orbital plane.

### 1. NR frame$(\hat{e}_x, \hat{e}_y, \hat{e}_z)$

This is just a generic coordinate system, without any regards to the concrete binary motion. Generic coordinate systems occur in numerical relativity, where the coordinates are chosen through same gauge-conditions, and the binary is evolving from some initial data. At some later time therefore, the coordinates will not have any particular, controlled properties.

The Cartesian basis-vectors are denoted

$$\hat{e}_x, \hat{e}_y, \hat{e}_z. \tag{14}$$

From these, spherical basis-vectors can be computed as usual

$$\hat{r} = \cos\phi \sin\theta \ \hat{e}_x + \sin\phi \sin\theta \ \hat{e}_y + \cos\theta \ \hat{e}_z \tag{15a}$$

$$\hat{\theta} = \cos\phi \cos\theta \ \hat{e}_x + \sin\phi \cos\theta \ \hat{e}_y - \sin\theta \ \hat{e}_z \tag{15b}$$

$$\hat{\phi} = -\sin\phi \ \hat{e}_x + \cos\phi \ \hat{e}_y. \tag{15c}$$

NR data is assumed to be represented by spherical-harmonic modes of the NR coordinates, as defined in the NINJA data-formats document [10]:

$$h_+^{\rm NR}(t_{\rm GW}) = \frac{1}{2}\left(\hat{\theta}_i\hat{\theta}_j - \hat{\phi}_i\hat{\phi}_j\right) h^{ij}(t_{\rm GW}) \tag{16a}$$

$$h_\times^{\rm NR}(t_{\rm GW}) = \frac{1}{2}\left(\hat{\theta}_i\hat{\phi}_j + \hat{\phi}_i\hat{\theta}_j\right) h^{ij}(t_{\rm GW}), \tag{16b}$$

and

$$\sum_{\ell=2}^{\infty}\sum_{m=-\ell}^{\ell} H_{\ell m}(t_{\rm GW})\ {}^{-2}Y_{\ell m}(\theta,\phi) = \frac{r}{M}\Big(h_+^{\rm NR}(t_{\rm GW}) - i h_\times^{\rm NR}(t_{\rm GW})\Big). \tag{17}$$

As explained in Sec. II, the data-files are assumed to contain a compressed time-series of amplitudes and phases per Eq. (3). Given an emission direction $(\theta, \phi)$, Eq. (17) yields the GW modes $h_+^{\rm NR}(t_{\rm GW})$ and $h_\times^{\rm NR}(t_{\rm GW})$, according to the convention Eqs. (16).

The gravitational wave data are given in a time-coordinate $t_{\rm GW}$ of observers at large distance.

Let us now turn to a description of the dynamics of the two bodies. Numerical relativity defines a variety of vectors, the combination of which defines the instantaneous state of the two bodies. These are: The dimensionless spin vectors of the two bodies,

$$\vec{\chi}_1(t), \vec{\chi}_2(t); \tag{18}$$

the direction from body 2 to body 1,

$$\hat{n}(t); \tag{19}$$

the direction of the Newtonian orbital angular momentum,

$$\hat{L}(t). \tag{20}$$

We do **not** make any assumption about the relation of the NR-orbital angular momentum vector $\hat{L}$ relative to the NR coordinates[16]. One can further define an orbital frequency

$$\vec{\Omega}(t) = \hat{n}(t) \times \frac{d\hat{n}(t)}{dt}. \tag{21}$$

Equations (18)–(21) are defined in the strong-field region near the black holes, and are given as a function of the time-coordinate $t$ employed by the NR simulation in the strong field regime.

The time-coordinates $t_{\rm GW}$ and $t$ are defined at different regions of the space-time (far-zone vs. near-zone). Sometimes it is desirable to relate these two time-coordinates to each other, e.g. to make a statement about the dynamical state of the binary at the time "when a certain GW-frequency is reached". Any relation between $t_{\rm GW}$ and $t$ is ambiguous. Often NR simulations employ an approximation of "retarded time", i.e.

$$t_{\rm GW} = t\Big|_{\rm retarted}. \tag{22}$$

However, in a dynamical space-time with black holes, two difficulties arise: First, the horizons are causally disconnected from future null infinity, so there are no outgoing null-rays that connect the horizons to the wave-zone. Second, when integrating null-rays "slightly outside" the horizons, the time-delay to infinity will depend on precisely where the integration was started, as well as on the initial direction of the null-ray. Therefore, any relation between $t_{\rm GW}$ and $t$ should be viewed as approximate. One should further assume that different NR groups may use different definitions of $t_{\rm GW}$ in the data they compute. For instance, `SpEC` reports GW-waveforms extracted at *finite-radius* in terms of the NR coordinate, $t_{\rm GW} = t$, whereas extrapolated waveforms are reported using a retarded time-coordinate with a correction of the rate of flow of time, see Equations (7), (14a) and (14b) of Boyle & Mroue [15].

### 2. *LAL Source-Frame* $(\hat{x}, \hat{y}, \hat{z})$

The LAL source-frame $(\hat{x}, \hat{y}, \hat{z})$ is defined as follows:

1. The $\hat{z}$-axis points along the orbital angular momentum of the binary,

$$\hat{z} \overset{\rm ref}{=} \hat{L}. \tag{23}$$

2. The $\hat{x}$-axis points along the vector $\hat{n}$ pointing from the second to the first body,

$$\hat{x} \overset{\rm ref}{=} \hat{n}. \tag{24}$$

3. The third vector $\hat{y} = \hat{z} \times \hat{x}$ completes the triad.

The symbol $\overset{\rm ref}{=}$ indicates that the respective equation is only required at the reference epoch. The reference epoch can be unambiguously specified by a reference-time $t_{\rm ref}$. One can also specify a reference *orbital* frequency $\Omega_{\rm ref}$, and infer $t_{\rm ref}$ via Eq. (21), $\Omega(t_{\rm ref}) = \Omega_{\rm ref}$. If the reference epoch is desired to be specified in terms of a *gravitational wave*–frequency, then the gravitational wave time $t_{\rm GW}$ needs to be related to the time-coordinate of the black hole dynamics, $t$. As discussed in the context of Eq. (22), such an identification is ambiguous and holds only approximately.

Equations (23) and (24) define the source-frame at the reference epoch only. They are chosen such that the spin-components (`Sx1, Sy1, Sz1`) and (`S2x, S2y, S2z`) have coordinate-invariant meaning: `Sx1` is the projection of $\vec{\chi}_1$ onto $\hat{n}$, `Sz1` is the projection of $\vec{\chi}_1$ onto $\hat{L}$, etc.

The source-frame does not rotate as the binary evolves. Specifically, the rotation between source-frame and wave-frame described below is constant in time.

A different reference epoch would lead to a different source-frame, related by some rotation. If the reference epoch is shifted by a small amount (comparable to the orbital time-scale), $\hat{x}$ and $\hat{y}$ would rotate with the binary. On the precession time-scale $\hat{z}$ would change.

The source frame has no deep intrinsic, geometric significance. It is merely a vehicle to describe the spin-projections onto intrinsic geometric directions, and the basis-vectors $(\hat{x}, \hat{y}, \hat{z})$ will be convenient when writing down the transformation to the wave-frame.

### 3. *Intrisinc parameters of a binary*

Given a reference epoch, a binary is specified by the following ten numbers:

- Two masses $m_1$, $m_2$.
- Two spin-vectors $\vec{\chi}_1$, $\vec{\chi}_2$, specified through the projections of the spin-vectors onto $\hat{L}$, $\hat{n}$ and the third basis-vector (at reference time).
- Eccentricity $e$.
- Mean anomaly $\delta$. The mean anomaly is defined in terms of the current time $t$, the time of last periapsis passage $T_{\text{previous}}$, and the time of next periapsis passage $T_{\text{next}}$:

$$\delta = 2\pi \frac{t - T_{\text{previous}}}{T_{\text{next}} - T_{\text{previous}}}. \tag{25}$$

Eccentricity should be defined somehow “near” the reference epoch. A precise definition (if possible at all) is left to the future. For low eccentricity orbits, there is a competition between radiation reaction driven inspiral (which results in a slightly negative average radial velocity), and the oscillatory radial motion due to eccentricity. For sufficiently small eccentricity, minima in separation may no longer exist. If mean anomaly is still needed despite the quite low eccentricity in such cases, one will have to define periapsis as minimum of separation compared to a fiducial smooth inspiral trajectory.

### 4. *Wave-Frame* $(\hat{X}, \hat{Y}, \hat{Z})$

The wave-frame is adopted to the direction of the observer (i.e. Earth), such that its $\hat{Z}$-axis points toward the observer. $\hat{X}$ and $\hat{Y}$ represent basis-vectors orthogonal to the line-of-sight, i.e. they span the plane of the sky.

The wave-frame is completely specified by three angles:

1. The angle $\Phi$ between line of ascending node and $\hat{n}$ (at the reference time).
2. The inclination $\iota$, i.e. the angle between orbital angular momentum $\hat{L} = \hat{z}$ and the line-of-sight $\hat{Z}$.
3. The angle $\Omega$ between the $\hat{X}$-axis and the line of ascending node (at the reference time).

These three angles happen to be the Euler angles of the rotation from wave-frame to source-frame.

The gravitational wave modes are defined as

$$h_{+}^{\mathrm{W}} = \frac{1}{2}\left(\hat{X}_i\hat{X}_j - \hat{Y}_i\hat{Y}_j\right)h^{ij}, \tag{26a}$$

$$h_{\times}^{\mathrm{W}} = \frac{1}{2}\left(\hat{X}_i\hat{Y}_j + \hat{Y}_i\hat{X}_j\right)h^{ij}, \tag{26b}$$

where the superscript ’W’ indicates the LAL wave-frame. The angle $\Omega$ merely rotates $\hat{X}$ and $\hat{Y}$ into each other. By definition Eq. (26), this merely rotates the polarization of the GW modes, and so $\Omega$ is fully degenerate with the GW polarization.

Note that the phase-angle $\Phi$ is specified without regard to the location of periapsis of the binary (this differs from the previous LAL convention). This new definition decouples the specification of periapsis location and the specification of orbital phase, and avoids the ambiguity that would arise in the zero-eccentricity limit of definitions involving periapsis. Indeed, for fixed $\Phi$ and fixed $\delta$, as eccentricity approaches zero, the waveforms will approach the identical circular form, independent of the value of $\delta$.

## B. Relationship between frames

### 1. From source-frame to wave-frame

The relation between source-frame and wave-frame can be easily derived following the three rotations that rotate one frame into the other: Beginning in the source-frame, $\hat{x}, \hat{y}, \hat{z}$, we first apply the rotation by $\Phi$ around $\hat{z}$, which yields a frame with basis-vectors $\hat{p}, \hat{q}, \hat{z}$:

$$\hat{p} = \cos\Phi\,\hat{x} - \sin\Phi\,\hat{y}, \tag{27a}$$
$$\hat{q} = \sin\Phi\,\hat{x} + \cos\Phi\,\hat{y}, \tag{27b}$$
$$\hat{z} = \hat{z}. \tag{27c}$$

The inverse rotation is

$$\hat{x} = \quad \cos\Phi\,\hat{p} + \sin\Phi\,\hat{q}, \tag{28a}$$
$$\hat{y} = -\sin\Phi\,\hat{p} + \cos\Phi\,\hat{q}, \tag{28b}$$
$$\hat{z} = \hat{z}. \tag{28c}$$

The vectors $\hat{p}, \hat{q}$ form an orthonormal basis of the $\hat{x}$-$\hat{y}$ plane, such that $\hat{p}$ points in the direction of ascending node.

Next, we rotate around the line of ascending node by the inclination $\iota$, resulting in basis-vectors

$$\hat{P} = \hat{p}, \tag{29a}$$
$$\hat{Q} = \cos\iota\,\hat{q} - \sin\iota\,\hat{z}, \tag{29b}$$
$$\hat{Z} = \sin\iota\,\hat{q} + \cos\iota\,\hat{z}. \tag{29c}$$

$\hat{P}$ and $\hat{Q}$ form an orthonormal basis of the $\hat{X}$-$\hat{Y}$ plane with $\hat{P}$ pointing in the direction of ascending node. The inverse rotation is

$$\hat{p} = \hat{P}, \tag{30a}$$
$$\hat{q} = \quad \cos\iota\,\hat{Q} + \sin\iota\,\hat{Z}, \tag{30b}$$
$$\hat{z} = -\sin\iota\,\hat{Q} + \cos\iota\,\hat{Z}. \tag{30c}$$

The final rotation rotates $\hat{P}$ and $\hat{Q}$ around $\hat{Z}$ into $\hat{X}, \hat{Y}$:

$$\hat{X} = \cos\Omega\,\hat{P} - \sin\Omega\,\hat{Q}, \tag{31a}$$
$$\hat{Y} = \sin\Omega\,\hat{P} + \cos\Omega\,\hat{Q}, \tag{31b}$$
$$\hat{Z} = \hat{Z}. \tag{31c}$$

The inverse rotation is

$$\hat{P} = \quad \cos\Omega\,\hat{X} + \sin\Omega\,\hat{Y}, \tag{32a}$$
$$\hat{Q} = -\sin\Omega\,\hat{X} + \cos\Omega\,\hat{Y}, \tag{32b}$$
$$\hat{Z} = \hat{Z}. \tag{32c}$$

Substituting Eqs. (27), (29) and (31) into each other, one obtains the entire transformation:

$$\begin{aligned}
\hat{X} =& (\cos\Omega\cos\Phi - \sin\Omega\cos\iota\sin\Phi)\,\hat{x} \\
&+ (-\cos\Omega\sin\Phi - \sin\Omega\cos\iota\cos\Phi)\,\hat{y} \\
&+ \sin\Omega\sin\iota\,\hat{z}, && (33a)\\
\hat{Y} =& (\sin\Omega\cos\Phi + \cos\Omega\cos\iota\sin\Phi)\,\hat{x} \\
&+ (-\sin\Omega\sin\Phi + \cos\Omega\cos\iota\cos\Phi)\,\hat{y} \\
&- \cos\Omega\sin\iota\,\hat{z}, && (33b)\\
\hat{Z} =& \sin\iota\sin\Phi\hat{x} + \sin\iota\cos\Phi\hat{y} + \cos\iota\,\hat{z}. && (33c)
\end{aligned}$$

The inverse transformation is obtained from Eqs. (28), (30) and (32):

$$\begin{aligned}
\hat{x} =& (\cos\Omega\cos\Phi - \sin\Omega\cos\iota\sin\Phi)\,\hat{X} \\
&+ (\sin\Omega\cos\Phi + \cos\Omega\cos\iota\sin\Phi)\,\hat{Y} \\
&+ \sin\iota\sin\Phi\hat{Z}, && (34a)\\
\hat{y} =& (-\cos\Omega\sin\Phi - \sin\Omega\cos\iota\cos\Phi)\,\hat{X} \\
&+ (-\sin\Omega\sin\Phi + \cos\Omega\cos\iota\cos\Phi)\,\hat{Y} \\
&+ \sin\iota\cos\Phi\hat{Z}, && (34b)\\
\hat{z} =& \sin\Omega\sin\iota\,\hat{X} - \cos\Omega\sin\iota\,\hat{Y} + \cos\iota\,\hat{Z}. && (34c)
\end{aligned}$$

Equation (34c) shows that $\Omega$ determines the direction of $\hat{L} = \hat{z}$ on the plane of the sky. For $\Omega = 0$, $\hat{L}$ lies in the $\hat{Y}$-$\hat{Z}$ plane, and for $\Omega = \pi/2$, it lies in the $\hat{X}$-$\hat{Z}$ plane.

This latter choice ($\Omega = \pi/2$) is already respected by many LAL waveform models. Therefore, in the absence of a reason to do otherwise, all waveform models should default to $\Omega = \pi/2$.

With this recommended default $\Omega = \pi/2$, Eqs. (33) simplify to

$$\begin{aligned}
\hat{X} &= -\cos\iota\sin\Phi\hat{x} - \cos\iota\cos\Phi\hat{y} + \sin\iota\,\hat{z}, && (35a)\\
\hat{Y} &= \cos\Phi\hat{x} - \sin\Phi\hat{y}, && (35b)\\
\hat{Z} &= \sin\iota\sin\Phi\hat{x} + \sin\iota\cos\Phi\hat{y} + \cos\iota\,\hat{z}. && (35c)
\end{aligned}$$

The inverse transformation (34) simplifies to

$$\begin{aligned}
\hat{x} &= -\cos\iota\sin\Phi\hat{X} + \cos\Phi\hat{Y} + \sin\iota\sin\Phi\hat{Z}, && (36a)\\
\hat{y} &= -\cos\iota\cos\Phi\hat{X} - \sin\Phi\hat{Y} + \sin\iota\cos\Phi\hat{Z}, && (36b)\\
\hat{z} &= \sin\iota\,\hat{X} + \cos\iota\,\hat{Z}. && (36c)
\end{aligned}$$

### 2. *From NR-frame to wave-frame*

Equations (33) and (35) are of particular importance. By definition, the source-frame basis-vectors are trivially related to vectorial quantities of the CBC dynamics:

$$\begin{aligned}
\hat{x} &\overset{\text{ref}}{=} \hat{n}, && (37a)\\
\hat{y} &\overset{\text{ref}}{=} (\hat{L} \times \hat{n}), && (37b)\\
\hat{z} &\overset{\text{ref}}{=} \hat{L}. && (37c)
\end{aligned}$$

Therefore, if the dynamics vectors $\hat{L}$ and $\hat{n}$ are known in any coordinate system —e.g., the NR coordinates— then Eqs. (33) yield the wave-frame basis-vectors in those coordinates. Specifically, for $\Omega = \pi/2$, Eqs. (35) yield

$$\hat{X} \overset{\text{ref}}{=} -\cos\iota\,\left[\sin\Phi\,\hat{n} + \cos\Phi\,\hat{L}\times\hat{n}\right] + \sin\iota\,\hat{L}, \tag{38a}$$

$$\hat{Y} \overset{\text{ref}}{=} \cos\Phi\,\hat{n} - \sin\Phi\,\hat{L}\times\hat{n}, \tag{38b}$$

$$\hat{Z} \overset{\text{ref}}{=} \sin i\left[\sin\Phi\,\hat{n} + \cos\Phi\,\hat{L}\times\hat{n}\right] + \cos i\,\hat{L}. \tag{38c}$$

Let us consider next the transformation of the GW strain polarizations from the generic (NR) frame Eq. (15) to the wave-frame. $\hat{X}$ and $\hat{Y}$ are orthogonal to the direction of propagation $\hat{Z} = \hat{r}$ of the gravitational wave. Therefore, $(\hat{\theta}, \hat{\phi})$ can be rotated into $(\hat{X}, \hat{Y})$ through a rotation by an angle $\alpha$:

$$\hat{X} = \cos\alpha\,\hat{\theta} - \sin\alpha\,\hat{\phi}, \tag{39a}$$

$$\hat{Y} = \sin\alpha\,\hat{\theta} + \cos\alpha\,\hat{\phi}. \tag{39b}$$

Substituting Eqs. (39) into Eqs. (26) we can compute $h_+^{\mathrm{W}}$ and $h_\times^{\mathrm{W}}$ in terms of $(\hat{\theta}, \hat{\phi})$. Comparing further with Eqs. (16), we find:

$$h_+^{\mathrm{W}}(t_{\mathrm{GW}}) = \cos(2\alpha)\, h_+^{\mathrm{NR}}(t_{\mathrm{GW}}) - \sin(2\alpha)\, h_\times^{\mathrm{NR}}(t_{\mathrm{GW}}), \tag{40a}$$

$$h_\times^{\mathrm{W}}(t_{\mathrm{GW}}) = \sin(2\alpha)\, h_+^{\mathrm{NR}}(t_{\mathrm{GW}}) + \cos(2\alpha)\, h_\times^{\mathrm{NR}}(t_{\mathrm{GW}}). \tag{40b}$$

Not surprising, the GW polarizations in the wave-frame are obtained from those in the NR-frame by a rotation of $2\alpha$.

The wave-frame $\hat{X}, \hat{Y}$ depend on $\Omega$, and therefore $h_+^{\mathrm{W}}$ and $h_\times^{\mathrm{W}}$ also depend on $\Omega$. We can make this dependence explicit by resorting to the intermediate vectors $\hat{P}$ and $\hat{Q}$. These are also orthogonal to $\hat{Z}$, therefore they, too, can be obtained from $\hat{\theta}$ and $\hat{\phi}$ by a rotation:

$$\hat{P} = \cos\alpha'\,\hat{\theta} - \sin\alpha'\,\hat{\phi}, \tag{41a}$$

$$\hat{Q} = \sin\alpha'\,\hat{\theta} + \cos\alpha'\,\hat{\phi}. \tag{41b}$$

However, $\hat{P}$ and $\hat{Q}$ are independent of $\Omega$ and therefore, $\alpha'$ is independent of $\Omega$. Because $(\hat{X}, \hat{Y})$ are rotated by $\Omega$ relative to $(\hat{P}, \hat{Q})$, we have

$$\alpha = \Omega + \alpha'. \tag{42}$$

Because rotations add, we can therefore write

$$\begin{pmatrix} h_+^{\mathrm{W}} \\ h_\times^{\mathrm{W}} \end{pmatrix} = \mathbf{R}_{2\Omega}\,\mathbf{R}_{2\alpha'} \begin{pmatrix} h_+^{\mathrm{NR}} \\ h_\times^{\mathrm{NR}} \end{pmatrix}, \tag{43}$$

where $\mathbf{R}_\beta$ denotes a 2x2 rotation matrix,

$$\mathbf{R}_\beta = \begin{pmatrix} \cos\beta & -\sin\beta \\ \sin\beta & \cos\beta \end{pmatrix}. \tag{44}$$

Equation (43) thus implies that the waveform-modes are obtained from the NR-polarizations by (i) applying an $\Omega$-independent rotation by $2\alpha'$; followed by (ii) a rotation by $2\Omega$.

For $\Omega = \pi/2$, we have $\mathbf{R}_\pi = -\mathbf{1}$. Therefore, between $\Omega = 0$ [inclination rotated about X-axis] and $\Omega = \pi/2$ [inclination rotated about Y-axis], the waveform polarization pick up precisely an overall minus-sign.

### C. Computing GW polarizations in the LAL wave-frame

Let us finally write down explicit instructions of how to obtain GW polarizations in LAL convention, given NR waveform data. Given parameters `i, phiRef` passed into `XLALSimInspiralChooseTDWaveform`, proceed as follows:

1. Define $\Phi = \text{phiRef}$, $\iota = \texttt{i}$.
2. Compute $\hat{Z}_{\text{ref}}$ at the reference time by evaluating Eq. (38c).
3. Because $\hat{Z}$ points in the direction of emission of the gravitational wave, we must have

$$\hat{Z}_{\text{ref}} \overset{\text{ref}}{=} \begin{pmatrix} \cos\phi\sin\theta \\ \sin\phi\sin\theta \\ \cos\theta \end{pmatrix}. \tag{45}$$

   From this equality, read off $(\theta,\phi)$. Then compute the NR-basis vectors $\hat{\theta},\hat{\phi}$ from Eqs. (15).

4. If $\Omega = \pi/2$, compute $\sin\alpha$ and $\cos\alpha$ by taking inner products of Eq. (39b) with $\hat{\theta}$ and $\hat{\phi}$:

$$\sin\alpha = \cos\Phi\,\hat{n}\cdot\hat{\theta} - \sin\Phi\,(\hat{L}\times\hat{n})\cdot\hat{\theta}, \tag{46a}$$

$$\cos\alpha = \cos\Phi\,\hat{n}\cdot\hat{\phi} - \sin\Phi\,(\hat{L}\times\hat{n})\cdot\hat{\phi}. \tag{46b}$$

   If $\Omega \neq \pi/2$, compute instead inner products based on Eqs. (33).

5. Substitute $(\theta,\phi)$ into Eqs. (17) and (16) to compute $h_+^{\text{NR}}(t_{\text{GW}})$ and $h_\times^{\text{NR}}(t_{\text{GW}})$.
6. Compute $\cos 2\alpha = \cos^2\alpha - \sin^2\alpha$ and $\sin 2\alpha = 2\cos\alpha\sin\alpha$. Substitute into Eqs. (40) to compute $h_+^{\text{W}}(t_{\text{GW}})$ and $h_\times^{\text{W}}(t_{\text{GW}})$.

#### *1. Evaluate spin-consistency in LAL source-frame*

The parameters `S1x, S1y, S1z, S2x, S2y,S2z` passed into `LALSimInspiralChooseTDwaveform` are supposed to be the LAL source-frame parameters, i.e. these parameters should simply be the projections of $\vec{\chi}_{1,2}$ onto the source-frame basis-vectors $(\hat{x},\hat{y},\hat{z})$. Substituting Eqs. (23) and (24), one arrives at the following consistency conditions:

$$\texttt{S1x} \overset{\text{ref}}{=} \vec{\chi}_1\cdot\hat{n}, \tag{47a}$$

$$\texttt{S1y} \overset{\text{ref}}{=} \vec{\chi}_1\cdot(\hat{L}\times\hat{n}), \tag{47b}$$

$$\texttt{S1z} \overset{\text{ref}}{=} \vec{\chi}_1\cdot\hat{L}. \tag{47c}$$

The conditions for body 2 are obtained by $1 \leftrightarrow 2$.

## V. DISCUSSION

With this new infrastructure it is very easy and much less memory intensive to use NR waveforms directly for data-analysis applications. The "NR_hdf5" approximant works much the same as any other approximant in lalsimulation, but there are a few important differences. The user must supply the location of the HDF5 file, a functionality which was already implemented for NINJA, but was not previously used in lalsimulation. Secondly, the user must be careful to supply the mass ratio and spin values that are consistent with the NR files, and the spin values have to be specified in the LAL source frame (see Eqn. (47)).

The current implementation still suffers from a few caveats and drawbacks. As opposed to the continuous waveform approximants, at the moment the metadata are only referring to the beginning of the waveform and not some reference time, which can be chosen freely. While this is not a problem for aligned-spin binaries, this is a big concern for precessing simulations since various quantities, in particular the spins and the orbital angular momentum, are time-dependent. To fully integrate this desired freedom, additional information needs to be incorporated into the HDF5 files and the waveform generator functions accordingly. Specifically, one needs the time-series of the vectors determining the geometry of the binary: $\hat{L}(t), \hat{n}(t), \chi_{(}t), \chi_2(t)$. Given these time-series, one can interpolate these four vectors to any reference epoch, and then apply the frame transformations at this reference epoch.

### Acknowledgements

We are grateful to Mark Hannam for many useful discussions and comments throughout the code review. We also thank Kent Blackburn and James Healy for providing useful comments on the manuscript. Many thanks for discussions regarding the frame coordinate transformations to Stas Babak, Jolien Creighton, Michael Pürrer and Riccardo Sturani.

[1] B. P. Abbott et al. (Virgo, LIGO Scientific), Phys. Rev. Lett. **116**, 061102 (2016), 1602.03837.

[2] L. Blanchet, Living Reviews in Relativity **17** (2014), URL `http://www.livingreviews.org/lrr-2014-2`.

[3] J. Centrella, J. G. Baker, B. J. Kelly, and J. R. van Meter, Rev. Mod. Phys. **82**, 3069 (2010), 1010.5260.

[4] F. Ohme, Class. Quant. Grav. **29**, 124002 (2012), 1111.3737.

[5] S. Khan, S. Husa, M. Hannam, F. Ohme, M. Pürrer, X. Jimnez Forteza, and A. Boh, Phys. Rev. **D93**, 044007 (2016), 1508.07253.

[6] A. Taracchini et al., Phys. Rev. **D89**, 061502 (2014), 1311.2544.

[7] J. Healy, C. O. Lousto, and Y. Zlochower, Phys. Rev. **D90**, 104004 (2014), 1406.7295.

[8] B. Aylott et al., Class. Quant. Grav. **26**, 165008 (2009), 0901.4399.

[9] J. Aasi et al. (VIRGO, LIGO Scientific, NINJA-2), Class. Quant. Grav. **31**, 115004 (2014), 1401.0939.

[10] D. Brown, S. Fairhurst, B. Krishnan, R. A. Mercer, R. K. Kopparapu, L. Santamaria, and J. T. Whelan (2007), 0709.0093.

[11] C. R. Galley and P. Schmidt, in prep. (2016), URL `https://dcc.ligo.org/LIGO-P1600064`.

[12] C. R. Galley, *romSpline*, URL `https://bitbucket.org/chadgalley/romspline`.

[13] T. Dal Canton et al., Phys. Rev. **D90**, 082004 (2014), 1405.6731.

[14] A. H. Mroue et al., Phys. Rev. Lett. **111**, 241104 (2013), 1304.6077.

[15] M. Boyle and A. H. Mroue, Phys. Rev. **D80**, 124045 (2009), 0905.3177.

[16] The customary choice of many NR groups is to *start* NR simulations with $\hat{L} = \hat{e}_z$. Because of junk-radiation and precession effects, $\hat{L}(t)$ will deviate from $\hat{e}_z$. This deviation generally will be very small for aligned-spin BBH systems, but may become significant for precessing systems.