LISA Data Challenge Manual

N/Ref :	LISA-LCST-SGS-MAN-001
Title	**LISA Data Challenge Manual**
Abstract	Users and developpers manual of the LISA Data Challenge (LDC)

	Name	**Date**	**Signature**
Prepared by	LDC WG	2020/07/21	
Checked by			
Checked by (QA)			
Approved by			

Document Change Record

Contributor List

Author's name	Institute	Location
Babak Stas	APC	Paris
Petiteau Antoine	APC	Paris
...		

Ver.	Date	Author	Description	Pages
0.1	2018-01-12	S. Babak, A. Petiteau	Initial version	all
0.1	2020/07/21	S. Babak (APC)	Current Version	

Distribution list

Recipient	Restricted	Not restricted
LISA Consortium		✗
LDC Participants		✗

Contents

1 Acronyms and Glossary

1.1 Acronyms

BH black hole

EMRI extreme mass-ratio inspiral

FAQ Frequently Asked Questions

GW Gravitational Wave

LDC LISA Data Challenge

LISA Laser Interferometer Space Antenna

MBHB Massive Black Hole Binary

PSD Power Spectral Density

SGWB Stochastic Gravitational Wave Background

SNR Signal-to-Noise Ratio

1.2 Glossary

2 Purpose and Scope

Laser Interferometer Space Antenna (LISA)

It is more than just fun, it is really critical for the LISA project!

3 Short description of Data Releases

This section collects the data releases. Each data release might contain multiple data sets united by a common purpose. Once you define the well posed question which you want to answer, you decided which sources you want to include in the data and if you want a simple or complex data. These define the number of data sets. The names of the data releases should reflect the purpose/project, though it might have a nick-name (just don't be rude, be funny). Each data set contains X, Y, Z TDI combinations (for the fractional frequency) in time domain.

3.1 Training data release "Radler"

This is the first data release after a long break. The main purpose if this data release is to build the LDC data production pipeline which eventually will be a part of a more complext end-to-end simulator. This challenge also aims at testing the collaboration tools (git lab, wiki, zoom, data base with the web interface). We have adopted hdf5 as a main data format for this challenge and the hdf5 file is a central part of the pipeline, we start with generating the hdf5 file based on the request (description of the data set) from user. This hdf5 file is get updated as it propagates trough the pipeline.

Scientific aim of this data release is build the data analysis tools and prototype of the pipeline. The data sets will not be very sophisticated. The tools used for analysing these data sets will be used as tutorials and examples. So the data sets are as follows:

1. Massive Black Hole Binary (MBHB) source: The gaussian instrumental noise and a single Gravitational Wave (GW) signals from MBHB. The data will be generated in the time domain using LISACode (for the response). We use IMRPhenomD waveform which describes inspiral-merger-ringdown in the frequency domain for non-precessing systems. The spins of both black hole (BH)s are parallel to the orbital angular momentum. The signal contains only the dominant ($l = 2, m = \pm 2$) mode. It is generated $(h_+, h_\times)$ in frequency domain and Fourier transformed to the time domain, then passed through the LISACode. The Signal-to-Noise Ratio (SNR) will be 100-500. Duration of the signal 0.6-1.2 years. Time step is 10 sec. Duration of observation is assumed to be 41943040 seconds.

2. MBHB source: The gaussian instrumental noise and a single GW signal from MBHB. It will be similar to the first data set with the following exceptions: (i) The signal will be generated completely in the frequency domain, including the TDI response (approximation). This procedure is described in the section 6.2.1.

3. We might also release the data set with a single MBHB signal similar to above. The instrumental noise will be assumed gaussian but its level will be chosen uniform $U[1, 2]$ of the nominal value for each link. The complications of this data set is that we do not know the level of the noise in each link and one cannot easily construct the TDI combination A, E, T with uncorrelated noise.

4. extreme mass-ratio inspiral (EMRI) source: The gaussian noise plus one EMRI GW signal as was used in the old MLDC, this is the model described in [5]. This model is not a faithful representation of the expected GW signal but fast to produce. In the data analysis algorithm the participants should not rely strongly on the model for the detection purposes. The SNR of the signal will be in the range 40-70, duration 1-1.5 years. Time step is 15 sec. Duration of observation is assumed to be 62914560 seconds.

5. EMRI source. The gaussian noise plus one EMRI signal. This data set is similar to the above one, but the model for the GW signal is based on the AAK (augmented analytic kludge) suggested in [7].

6. Galactic binaries: The gaussian noise and GW signals from the population of Galactic white dwarf binaries. The population contains about 30 millions of binary systems. The waveform $(h_+, h_\times)$ is produced by Taylor expansion of the phase (up to first derivative in frequency) at the t_0 (beginning of observations). The response function is approximate and described in details [8]. Time step is 15 sec. Duration of observation is assumed to be 62914560 seconds.

   We have posted another data set which contains 10 verification binaries. Finding and characterizing those sources should be rather easy and could serve as an excersise for the undergraduate students.

7. Stellar mass black hole binaries: The gaussian noise plus the GW signal from the population of the stellar mass binary BHs. Those are the binaries simiar to those detected by LIGO-VIRGO. Some of those binaries will be detectable in the band of ground based detectors several years after being observed in LISA. The waveforms $h_+, h_\times$ are produced using

the time domain (IMRPhenomD) model. The waveform (including the TDI response) is completely produced in the frequency domain using the approach described in 6.2.1 and then transformed into time domain. The signal was not tapered, so Gibbs oscillations are present in the time domain data. In addition the time step (5 sec.) imposes the Nyquist frequency 0.1 Hz, so all signals are generated up to this frequency, as result some signals are terminated abruptly in time domain. We assume that this data set will be analyzed in the frequency domain. Duration of observation is assumed to be 83886080 seconds. The population contains 21721 black hole binaies.

8. Bright stellar mass black hole binaries: this data set is similar to the above but we have used only the signals which have the total SNR above 5.0 (against the instrumental noise!). Those binaries are identical in parameters as in the data set above. Those interested in the stochastic signal produced by SOBBH (stellar origin binary black holes) can subtract this signal from the data set with a whole population.

9. Stochastic GW signal. Here we will have a gaussian noise plus stochastic GW signal. The stochastic GW signal has flat (across the whole frequency band) spectrum. The stochastic signal is isotropic

10. Stochastic GW signal. This data set is similar to the above but the spectrum is more complicated and described in the section 6.9

All datasets are stored together with the metadata (used to produce it) in hdf5 files. In particular we store $h_+, h_\times$ if we have used LISACode to generate the dataset otherwise we store X, Y, Z TDI channels (for the fractional frequency measurements). In all datasets (besides 3rd) we have used the noise with the same properties in each link, so one can form noise uncorrelated TDIs by a linear combination:

$$A = (Z - X)/\sqrt{2} \quad E = (X - 2Y + Z)/\sqrt{6}, \quad T = (X + Y + Z)/\sqrt{3}.$$

Note that T combination is not sensitive to GW signal at low frequencies.

4 Quick start

For the quick start instruction, you can have a look at the `README.md` or directly at the home webpage of the project on gitlab (https://gitlab.in2p3.fr/stas/MLDC).

5 Installation

5.1 Docker for user

There is a ready-to-use docker image containing an installation of all the LDC tools. This image is automatically updated after each push passing the tests. To use the image you need Docker (https://www.docker.com/) installed on your system. Once installed Docker, you can use it by entering following commands in your terminal:

```
docker login gitlab-registry.in2p3.fr
docker pull gitlab-registry.in2p3.fr/stas/mldc:master
docker run -it -v PathMyLocalDir:/workspace gitlab-registry.in2p3.fr/stas/mldc:master
```

- `PathMyLocalDir` is the path to a local directory on your computer that is mirrored with the directory `workspace` on the docker instance. It is use to exchange files between docker and the rest of your computer.
- `master` is the branch used to build the image

If everything went well, the last line of your terminal should start with `bash-4.2#`. If it's the case you are in the docker machine contianing the LDC tools. You are in the `/workspace` directory on docker which is a mirror of your local directory `PathMyLocalDir`:

```
bash-4.2# pwd
/workspace
bash-4.2#
```

5.2 Docker for developer

There is docker image containing all the necessary packages to install and to run the LDC tools. To use the image you need Docker (https://www.docker.com/) installed on your system.

Once installed Docker, make sure you have `git` installed and then:

1. Clone the LDC git repository somewhere on your computer (`MYDIR/LDC`):

```
cd MYDIR/LDC
git clone git@gitlab.in2p3.fr:stas/MLDC.git
```

2. Start docker

```
docker login gitlab-registry.in2p3.fr
docker pull gitlab-registry.in2p3.fr/elisadpc/docker:ldc_env
docker run -it -v /PATH_TO_DIR_WITH_LDC:/workspace \
    gitlab-registry.in2p3.fr/elisadpc/docker:ldc_env
```

The last line of your terminal should start with `workspace#`: your terminal is in the docker instance which a standard unix environment (CentOS) and containing all the tools necessary for the LDC. The directory `workspace` on the docker instance is a mirror of your local directory `/PATH_TO_DIR_WITH_LDC` which should be the directory containing the `MLDC` directory cloned from git. If you type `ls` you should see the `MLDC` directory.

3. Install the LDC tools and run the tests:

```
cd MLDC
python master_install.py
export PYTHONPATH=/workspace/LDC/software/LDCpipeline/scripts/:$PYTHONPATH
export PATH=/workspace/LDC/software/LDCpipeline/scripts/:$PATH
python -m unittest discover
```

The two `export` lines are needed for indicating where are the function that some script are calling and that are tested. `/workspace/LDC` is the path to the LDC directory on the docker machine.

Now, you can make your change on the LDC scripts,etc and run the tools in the terminal

5.3 Manual installation

5.3.1 How to

1. Make sure you have all the dependencies (see section 5.3.2) installed.
2. Clone the repository:

```
git clone git@gitlab.in2p3.fr:stas/MLDC.git
```

3. Run the `master_install`:

```
cd MLDC
python master_install.py --prefix=WhereToInstall
```

5.3.2 Dependencies

You will need to install:

- c++ compiler
- `python`
- `gsl`
- `fftw`
- `cython`
- `swig`
- LISACode (https://gitlab.in2p3.fr/elisadpc/LISACode) (optional. if you want to apply response in the time domain)

6 Waveform inventory

6.1 Convention for binary systems

6.1.1 Source frame

We start by introducing a source frame defined by unit vectors $(\boldsymbol{x}_S, \boldsymbol{y}_S, \boldsymbol{z}_S)$. The orientation of this source frame with respect to the binary system considered is left to the waveform model, as different choices might be preferable for different systems. An example is given by the NR convention of [13, 12], where the unit separation vector is $\boldsymbol{n} = \boldsymbol{x}_S$ and the normal to the orbital plane is $\hat{\boldsymbol{L}}_N = \boldsymbol{z}_S$ at the start of the waveform.

In this source frame, we introduce standard spherical coordinates (θ_S, ϕ_S), and the associated spherical orthonormal basis vectors $(\boldsymbol{e}_r^S, \boldsymbol{e}_\theta^S, \boldsymbol{e}_\phi^S)$. The unit vector $\boldsymbol{k}$ defines the direction of propagation of the gravitational waves, from the source towards the observer. Expressed in spherical coordinates,

$$\boldsymbol{k} = \boldsymbol{e}_r^S = \{\sin\theta_S \cos\phi_S, \sin\theta_S \sin\phi_S, \cos\theta_S\}\,. \tag{1}$$

The explicit expressions for the other vectors of the spherical basis are

$$\boldsymbol{e}_\theta^S = \frac{\partial \boldsymbol{e}_r^S}{\partial \theta_S} = \{\cos\theta_S \cos\phi_S, \cos\theta_S \sin\phi_S, -\sin\theta_S\}\,, \tag{2a}$$

$$\boldsymbol{e}_\phi^S = \frac{1}{\sin\theta_S}\frac{\partial \boldsymbol{e}_r^S}{\partial \phi_S} = \{-\sin\phi_S, \cos\phi_S, 0\}\,. \tag{2b}$$

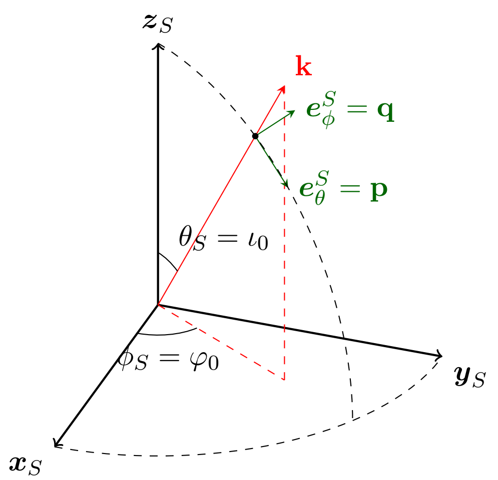

Figure 1: Source frame.

Next, introduce the polarization basis vectors:

$$\boldsymbol{p} = \boldsymbol{e}_\theta^S, \quad \boldsymbol{q} = \boldsymbol{e}_\phi^S \tag{3}$$

which form together with $\boldsymbol{k}$ the radiation frame or wave-frame: $(\boldsymbol{p}, \boldsymbol{q}, \boldsymbol{k})$. They can be defined from $\boldsymbol{z}_S$ only as

$$\boldsymbol{q} = \frac{\boldsymbol{z}_S \times \boldsymbol{k}}{|\boldsymbol{z}_S \times \boldsymbol{k}|}\,, \quad \boldsymbol{p} = \boldsymbol{q} \times \boldsymbol{k}. \tag{4}$$

The source frame and the polarization vectors are shown in the fig. 1. Defining the polarization tensors

$$\boldsymbol{e}_{ij}^{+} = (\boldsymbol{p}\otimes\boldsymbol{p} - \boldsymbol{q}\otimes\boldsymbol{q})_{ij}\,, \quad \boldsymbol{e}_{ij}^{\times} = (\boldsymbol{p}\otimes\boldsymbol{q} + \boldsymbol{q}\otimes\boldsymbol{p})_{ij}\,, \tag{5}$$

the GW strain in transverse-traceless gauge takes the form

$$h_{ij}^{\mathrm{TT}} = \boldsymbol{e}_{ij}^{+} h_{+} + \boldsymbol{e}_{ij}^{\times} h_{\times} \,. \tag{6}$$

This defines the polarizations h_+ and $h_\times$, functions of time. They are also given by the inverse relations

$$h_{+} = \frac{1}{2} h_{ij}^{\mathrm{TT}} \boldsymbol{e}_{ij}^{+} \,, \quad h_{\times} = \frac{1}{2} h_{ij}^{\mathrm{TT}} \boldsymbol{e}_{ij}^{\times} \,. \tag{7}$$

The terminology used in the code tries to stay close to usual names and notation. The 0 subscript reminds us that these angles are constants.

$$\text{Inclination :} \quad \iota_0 \equiv \theta_S \,, \tag{8}$$
$$\text{Observer phase :} \quad \varphi_0 \equiv \phi_S \,. \tag{9}$$

6.1.2 From the source frame to the SSB frame

We now introduce as a detector frame a Solar System Barycenter (SSB) frame $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$, based on the ecliptic plane. Note that although we parallel here the geometric definitions of `LAL` [1], in their case this detector frame is a different frame, geocentric and based on the celestial equator.

Like for the source frame, we introduce standard spherical coordinates in the SSB-frame (θ, ϕ), and the associated spherical orthonormal basis vectors $(\boldsymbol{e}_r, \boldsymbol{e}_\theta, \boldsymbol{e}_\phi)$. The position of the source in the sky will be parametrized by the ecliptice latitude $\beta = \pi/2 - \theta$ and the ecliptic longitude $\lambda = \phi$. The GW propagation vector $\boldsymbol{k}$ in spherical coordinates is now

$$\boldsymbol{k} = -\boldsymbol{e}_r = \{-\cos\beta\cos\lambda, -\cos\beta\sin\lambda, -\sin\beta\} \,. \tag{10}$$

Again, the explicit expressions for the other vectors of the spherical basis are

$$\boldsymbol{e}_\theta = \{\sin\beta\cos\lambda, \sin\beta\sin\lambda, -\cos\beta\} \,, \tag{11a}$$
$$\boldsymbol{e}_\phi = \{-\sin\lambda, \cos\lambda, 0\} \,. \tag{11b}$$

Introduce reference polarization vectors as

$$\boldsymbol{u} = -\boldsymbol{e}_\phi = \{\sin\lambda, -\cos\lambda, 0\} \,, \tag{12a}$$
$$\boldsymbol{v} = -\boldsymbol{e}_\theta = \{-\sin\beta\cos\lambda, -\sin\beta\sin\lambda, \cos\beta\} \,, \tag{12b}$$

so that $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{k})$ form a direct orthonormal triad. Equivalent expressions directly in terms of $\boldsymbol{k}$ and $\boldsymbol{z}$ are

$$\boldsymbol{u} = \frac{\boldsymbol{z} \times \boldsymbol{k}}{|\boldsymbol{z} \times \boldsymbol{k}|} \,, \quad \boldsymbol{v} = \boldsymbol{k} \times \boldsymbol{u}. \tag{13}$$

The SSB frame and its reference polarization vectors are given schematically in the fig 2.

The last degree of freedom between the frames corresponds to a rotation around the line of sight, and is represented by the polarization angle ψ. We define the polarization to be the angle of the the rotation around $\boldsymbol{k}$ that maps $\boldsymbol{u}$ to $\boldsymbol{p}$ (see figure 3):

$$\boldsymbol{p} = \boldsymbol{u}\cos\psi + \boldsymbol{v}\sin\psi, \tag{14a}$$
$$\boldsymbol{q} = -\boldsymbol{u}\sin\psi + \boldsymbol{v}\cos\psi. \tag{14b}$$

The polarization angle can be computed as[1]

$$\psi = \mathrm{arctan}_2 \left[\boldsymbol{p} \cdot \boldsymbol{u}, \boldsymbol{p} \cdot \boldsymbol{v}\right] \,. \tag{15}$$

[1] With the convention that $\mathrm{arctan}_2[x, y]$ is the polar angle of the point of coordinates (x, y). Note that the function `numpy.arctan2` uses a reverse ordering $[y, x]$.

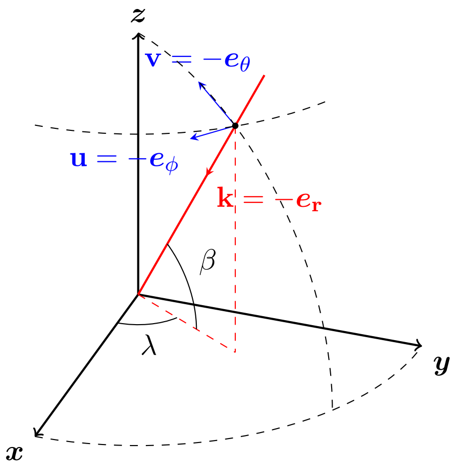

Figure 2: SSB frame

Given the direction of $z_S = \{\cos\phi_{zS}\sin\theta_{zS}, \sin\phi_{zS}\sin\theta_{zS}, \cos\theta_{zS}\}$ in SSB we can compute the inclination and polarization as

$$\iota_o = \arccos\left[-\cos\theta_{zS}\sin\beta - \cos\beta\sin\theta_{zS}\cos(\lambda - \phi_{zS})\right] \tag{16}$$

$$\tan\psi = \frac{-\sin\beta\sin\theta_{zS}\cos(\lambda-\phi_{zS}) + \cos\theta_{zS}\cos\beta}{\sin\theta_{zS}\sin(\lambda-\phi_{zS})} \tag{17}$$

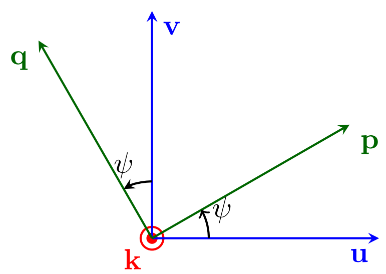

Figure 3: Polarization angle.

Defining as in (18) polarization basis tensors associated to $(\boldsymbol{u}, \boldsymbol{v})$ as

$$\epsilon^{+}_{ij} = (\boldsymbol{u}\otimes\boldsymbol{u} - \boldsymbol{v}\otimes\boldsymbol{v})_{ij}\,, \quad \epsilon^{\times}_{ij} = (\boldsymbol{u}\otimes\boldsymbol{v} + \boldsymbol{v}\otimes\boldsymbol{u})_{ij}\,, \tag{18}$$

the relation between the polarization tensors is

$$\boldsymbol{e}^{+} = \boldsymbol{\epsilon}^{+}\cos 2\psi + \boldsymbol{\epsilon}^{\times}\sin 2\psi, \tag{19a}$$

$$\boldsymbol{e}^{\times} = -\boldsymbol{\epsilon}^{+}\sin 2\psi + \boldsymbol{\epsilon}^{\times}\cos 2\psi. \tag{19b}$$

and corresponding representation of the strain in the SSB frame is

$$h^{SSB}_{ij} = (h_{+}\cos 2\psi - h_{\times}\sin 2\psi)\,\epsilon^{+}_{ij} + (h_{+}\sin 2\psi + h_{\times}\cos 2\psi)\,\epsilon^{\times}_{ij}. \tag{20}$$

The terminology used in the code is:

$$\text{Ecliptic longitude}: \quad \lambda \equiv \phi\,, \tag{21}$$

$$\text{Ecliptic latitude}: \quad \beta \equiv \pi/2 - \theta\,, \tag{22}$$

$$\text{Polarization}: \quad \psi\,. \tag{23}$$

6.1.3 Mode decomposition

Those two polarizations can be decomposed in the spin-weighted (-2) spherical harmonics:

$$h_+(t) - ih_\times(t) = \sum_{\ell=2}^{+\infty} \sum_{m=-\ell}^{\ell} h_{\ell m}(t)\,{}_{-2}Y_{\ell m}(\iota_0, \varphi_0)\,. \tag{24}$$

Our convention for the spin-weighted spherical harmonics is the same as the one used in `LAL` [1], in [2] and [6]:

$${}_{-2}Y_{\ell m}(\iota_0, \varphi_0) = \sqrt{\frac{2\ell+1}{4\pi}} d^\ell_{m,2}(\iota_0) e^{im\varphi_0}\,, \tag{25a}$$

$$d^\ell_{m,2}(\iota_0) = \sum_{k=k_1}^{k_2} \frac{(-1)^k}{k!} \frac{\sqrt{(\ell+m)!(\ell-m)!(\ell+2)!(\ell-2)!}}{(k-m+2)!(\ell+m-k)!(\ell-k-2)!} \left(\cos\frac{\iota_0}{2}\right)^{2\ell+m-2k-2} \left(\sin\frac{\iota_0}{2}\right)^{2k-m+2}\,, \tag{25b}$$

with $k_1 = \max(0, m-2)$ and $k_2 = \min(\ell+m, \ell-2)$. The polarizations can be expressed as

$$h_+ = \frac{1}{2} \sum_{\ell,m} \left({}_{-2}Y_{\ell m} h_{\ell,m} + {}_{-2}Y^*_{\ell,-m} h^*_{\ell,m} \right)\,, \tag{26a}$$

$$h_\times = \frac{i}{2} \sum_{\ell,m} \left({}_{-2}Y_{\ell m} h_{\ell,m} - {}_{-2}Y^*_{\ell,-m} h^*_{\ell,m} \right)\,, \tag{26b}$$

Note that our polarization vectors (4) differ from the PN convention of [6] by a rotation of $\pi/2$, which translates into an overall factor (-1) in the polarizations $h_{+,\times}$ and in the modes $h_{\ell m}$.

For non-precessing binary systems, with a fixed equatorial plane of orbit, an exact symmetry relation between modes holds:

$$h_{\ell,-m} = (-1)^\ell h^*_{\ell m}\,. \tag{27}$$

When this symmetry is verified, we can write

$$h_{+,\times} = \sum_{\ell,m} K^{+,\times}_{\ell m} h_{\ell m}\,, \tag{28}$$

with

$$K^+_{\ell m} = \frac{1}{2} \left({}_{-2}Y_{\ell m} + (-1)^\ell {}_{-2}Y^*_{\ell,-m} \right)\,, \tag{29a}$$

$$K^\times_{\ell m} = \frac{i}{2} \left({}_{-2}Y_{\ell m} - (-1)^\ell {}_{-2}Y^*_{\ell,-m} \right)\,. \tag{29b}$$

As an example of the connection between modes and polarizations, let us consider the emission from a non-precessing binary system on a circular orbit, at the leading post-Newtonian order. The source frame is such that $\boldsymbol{z}_S = \hat{\boldsymbol{L}}_N$ is the normal to the orbital plane, and Φ denotes the phase of the binary separation vector $\boldsymbol{y}_1 - \boldsymbol{y}_2 = r\boldsymbol{n}$ (with $\boldsymbol{y}_{1,2}$ the position of the two bodies) in the plane $(\boldsymbol{x}_S, \boldsymbol{y}_S)$. We also use the orbital frequency $\omega = \dot{\Phi}$.

At leading PN order, only the dominant $\{\ell, m\} = \{2, \pm 2\}$ mode enters. From Sec. (9.5) of [6], taking into account the sign change caused by the difference in the definition of $\boldsymbol{p}, \boldsymbol{q}$,

$$h_{22} = -\frac{2Gm\nu x}{Rc^2} \sqrt{\frac{16\pi}{5}} e^{-2i\Phi}\,, \tag{30a}$$

$$h_{2,-2} = h^*_{22}\,, \tag{30b}$$

where we introduced $m = m_1 + m_2$ the total mass, $\nu = m_1 m_2/m^2$ the symmetric mass ratio, $x = (Gm\omega/c^3)^{2/3}$ is the PN parameter and R is the distance to the source. The expressions of the spin-weighted spherical harmonics (25a) for $\{\ell, m\} = \{2, \pm 2\}$ are

$$
{}_{-2}Y_{22}(\iota_0, \varphi_0) = \frac{1}{2}\sqrt{\frac{5}{\pi}} \cos^4 \frac{\iota_0}{2} e^{2i\varphi_0} \,, \tag{31a}
$$

$$
{}_{-2}Y_{2,-2}(\iota_0, \varphi_0) = \frac{1}{2}\sqrt{\frac{5}{\pi}} \sin^4 \frac{\iota_0}{2} e^{-2i\varphi_0} \,. \tag{31b}
$$

Using the combinations (29),

$$
\begin{aligned}
h_+ &= K^+_{22} h_{22} + K^+_{2,-2} h_{2,-2} \\
&= -\frac{4Gm\nu x}{Rc^2} \frac{1 + \cos^2 \iota_0}{2} \cos\left(2(\Phi - \varphi_0)\right) \,, \qquad (32a) \\
h_\times &= K^\times_{22} h_{22} + K^\times_{2,-2} h_{2,-2} \\
&= -\frac{4Gm\nu x}{Rc^2} \cos \iota_0 \sin\left(2(\Phi - \varphi_0)\right) \,. \qquad (32b)
\end{aligned}
$$

6.1.4 Time and phase

6.1.5 Frequency domain

For the definitions of the Fourier transform (FT), we use the same convention as in [1], used also `numpy` (and up to normalization in `FFTW`): for a function of time $F(t)$

$$
\tilde{F}(f) = \int dt \; F(t) e^{-2i\pi f t} \,, \quad F(t) = \int df \; \tilde{f}(f) e^{2i\pi f t} \,. \tag{33}
$$

The discrete Fourier transform for N time samples F_j and frequency samples $\tilde{F}_k$ is then

$$
\tilde{F}_k = \sum_{j=0}^{N-1} F_j e^{-2i\pi \frac{jk}{N}} \,, \quad F_j = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{F}_k e^{2i\pi \frac{jk}{N}} \,. \tag{34}
$$

For a real time-domain signal,

$$
F(t) \in \mathbb{R} \implies \tilde{F}(-f) = \tilde{F}(f)^* \,, \tag{35}
$$

and we recall the following useful relation valid in general:

$$
\widetilde{F^*}(f) = \tilde{F}(-f)^* \,. \tag{36}
$$

For non-precessing systems, (27) implies

$$
\tilde{h}_{\ell,-m}(f) = (-1)^\ell \tilde{h}_{\ell m}(-f)^* \,. \tag{37}
$$

Since a given mode has a phase dependency $h_{\ell m} \propto \exp[-im\phi_{\rm orb}]$, with the orbital phase verifying $\dot{\phi}_{\rm orb} > 0$, an approximation often used for non-precessing systems (or in the precessing frame for a binary with misaligned spins) is

$$
\tilde{h}_{\ell m}(f) \simeq 0 \text{ for } m > 0 \,, f > 0 \,, \tag{38a}
$$

$$
\tilde{h}_{\ell m}(f) \simeq 0 \text{ for } m < 0 \,, f < 0 \,, \tag{38b}
$$

$$
\tilde{h}_{\ell 0}(f) \simeq 0 \,. \tag{38c}
$$

Note that in the Fourier convention (33), this approximation means that for positive frequencies $f > 0$ the mode $\tilde{h}_{2,-2}(f)$ has support while $\tilde{h}_{22}(f)$ is negligible[2]. When using the approximation (38), (39) becomes for $f > 0$

$$\tilde{h}_{+,\times}(f) = \sum_{\ell} \sum_{m<0} K_{\ell m}^{+,\times} \tilde{h}_{\ell m} \,. \tag{39}$$

6.1.6 Link to the LAL conventions

In conclusion of this section we explain of our conventions relate to those used in `LAL` (software library used by LIGO/Virgo collaborations [1]), which are standardized in the documents [13, 12]. First, in `LAL` the source frame is constructed by convention [13] setting the unit separation vector $\boldsymbol{n} = \boldsymbol{x}_S$ and the normal to the orbital plane $\hat{\boldsymbol{L}}_N = \boldsymbol{z}_S$, at the start of the waveform. In our case we have not specified how this source-frame is built, as this construction might differ for different physical systems (SMBHs, EMRIs, ...).

Translated in the notations we used above, the choice of polarization vectors for LAL is

$$\boldsymbol{p}_{\mathrm{LAL}} = -\boldsymbol{p} \,, \tag{40a}$$
$$\boldsymbol{q}_{\mathrm{LAL}} = -\boldsymbol{q} \,, \tag{40b}$$

which amounts to a rotation of π of the polarization basis. Because the polarizations $h_{+,\times}$ transform with $\cos 2\psi, \sin 2\psi$ under a change of polarization, this means that there is no difference here, provided the source-frame is identical:

$$h_{+}^{\mathrm{LAL}} = h_{+} \,, \tag{41a}$$
$$h_{\times}^{\mathrm{LAL}} = h_{\times} \,. \tag{41b}$$

The phase quantity used in `LAL` differs from our definition by

$$\Phi_{\mathrm{LAL}} = \frac{\pi}{2} - \varphi_0 \,. \tag{42}$$

Although the geometric definition of the reference polarization vectors $\boldsymbol{u}$, $\boldsymbol{v}$ is the same, the detector frame $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z})$ is different. In the case of `LAL`, it is a geocentric frame based on the celestial equator, while in our case it is the SSB frame based on the ecliptic plane. For the polarization, a difference of π in the definition also comes from (40):

$$\psi_{\mathrm{LAL}} \leftrightarrow \psi \ : \text{diff. of } \pi \text{ and different frame} \,. \tag{43}$$

For the sky position, `LAL` uses $\mathrm{ra}, \mathrm{dec}$ instead of our ecliptic λ, β:

$$(\alpha_{\mathrm{LAL}}, \beta_{\mathrm{LAL}}) \leftrightarrow (\lambda, \beta) \ : \ \text{different frame} \,. \tag{44}$$

We leave for future work this exact map, relating the SSB frame to the geocentric frame.

6.2 MBHB: Massive black hole binaries

6.2.1 IMRPhenomD

IMRPhenomD was introduced in [9] and we will not repeat here its description. Instead we will describe how the TDI data was produced. By construction the Fourier transformation of $h+, h_{\times}$ into time domain is such that the merger is around $t = 0$. We shift the waveform by T_c

[2] Note that because of this, some parts of the LDC code internally use the opposite sign convention in the exponentials of (33), so as to have support for the modes $m > 0$ for $f > 0$. This amounts to a mapping $f \leftrightarrow -f$, which can be undone at the end of the computation by conjugating the FT of the observables, which are real signals (see (35)).

(coalescence time). Then data is passed through the LISACode to apply the instrument response. For generation of MBHBs we have used the reference frequency to be 3mHz. For generation of SOBBH we have used reference time to be start of observation and the reference frequency is the instantaneous GW frequency at the beginning of observation. Location: software/Waveforms/MBH_IMR/IMRPhenomD/.

6.3 TDI response in frequency domain

We also could generate the X, Y, Z TDIs in the frequency domain (approximately). The key procedure is described in the document [11] and test/example python notebooks are provided in /MBH_IMR/IMRPhenomD/examples/. We generate the full waveform with response in the frequency domain, and Fourier transform it to the time domain. This was done for one data set containing MBHBs and for data sets with SOBBHs. The most complete code which uses the FD response is software/LDCpipeline/scripts/MakeSOBBH_FD.py, the MBHB uses basically the same code but with stencil order equal to zero and instead of f_0 uses t_c to determine the starting frequency.

6.4 MBHB parametrization.

Parameters for the source together with units are stored in the hdf5 file. In the table 1 we give parameters describing the MBHB signal.

Parameter	Description	units
β	EclipticLatitude	Radian
λ	EclipticLongitude	Radian
θ_{S1}	PolarAngleOfSpin1	Radian
θ_{S2}	PolarAngleOfSpin2	Radian
ϕ_{S1}	AzimuthalAngleOfSpin1	Radian
ϕ_{S2}	AzimuthalAngleOfSpin2	Radian
s_1	magnitude Spin1	MassSquared
s_2	magnitude Spin2	MassSquared
m_1	Mass1	SolarMass
m_2	Mass2	SolarMass
t_c	CoalescenceTime	Second
ϕ_{ref}	PhaseAtCoalescence	Radian
θ_L	InitialPolarAngleL	Radian
ϕ_L	InitialAzimuthalAngleL	Radian
-	Approximant	ModelName
Δt	Cadence	Seconds
z	Redshift	dimensionless
D_L	Distance (luminosity)	Gpc
T_{obs}	ObservationDuration	Seconds

Table 1: Parametrization used for MBHBs.

Let us also give some necessary explanations. We create those parmeters to cover also precessing binaries (eccentric binaries will require an extension of this table), therefore some parameters (ϕ_{S1}, ϕ_{S2}) are redundant. The spin used for the IMRPhenomD model are computed as $a_i = s_i \cos\theta_{Si}$. Masses quoted in the hdf5 files are always redshifted masses. The time of coalescence roughly corresponds to the merger of two BHs.

We have given redundant information: redshift z and the luminosity distance D_L are connected by adopted cosmological model (based on Planck2015 parameters).

6.5 Technical detail on the production of Radler data sets.

The script used to produce the MBHB data sets is software/LDCpipeline/scripts/ProduceMBHB_Radler.py and parameter file is software/LDCpipeline/scripts/Param_RadlerMBHB.txt. We use the source catalogues to select the source specified in the parameter file. In the next sections we will specify the conventions for TDI and the waveforms, those conventions are coded up as a class "LW" software/LDCpipeline/scripts/LW_simple.py. The comparison between three codes (LISACode, LW, and FD (frequency domain response) is given by running script software/LDCpipeline/scripts/MBHB_LW_FD_LC_SNR.py there we also compute the SNR of the signal in the data set. We have used LISACode as a reference and to have an agreement we had to modify the polarization angle as $\psi \to \psi_{FD} = \pi/2 - \psi$ and $\psi \to \psi_{LW} = \pi/2 + \psi$ for the "FD" and "LW" implementations. The command used to produce the data set:

```
$ python ProduceMBHB_Radler.py --verbose \
--seed=54321 --param=Param_RadlerMBHB.txt LDC1-1_MBHB_v1.hdf5
```

6.6 EMRI

6.6.1 Analytic Kludge model

The AK model used in the Radler challenge is based on [5] for generation of $h_+, h_\times$ in the SSB frame. The major modification is that we fix the number of harmonics (similar to how it was done in the old MLDC [3]). The TDI response is applied using LISACode.

We summarize in the table 2 the waveform parametrization. The Radler data set was produced using the script software/LDCpipeline/scripts/ProduceEMRI.py.

Description	Parameter	Notation	units
Sky position (SSB)	β	EclipticLatitude	Radian
Sky position (SSB)	λ	EclipticLongitude	Radian
Mass of SMBH	M	MassOfSMBH	SolarMass
Mass of compact object	μ	MassOfCompactObject	SolarMass
SMBH spin	S	SMBHspin	MassSquared
SMBH spin orient. (in SSB)	θ_K, ϕ_K	PolarAngleOfSpin	Radian
Radial orb. freq. ($t = 0$)	ν_0	InitialAzimuthalOrbitalFrequency	Hertz
Orb. mean anom. ($t = 0$)	Φ_0	InitialAzimuthalOrbitalPhase	Radian
Eccentricity ($t = 0$)	e_0	InitialEccentricity	1
Dir. of pericenter ($t = 0$)	$\tilde{\gamma}_0$	InitialTildeGamma	Radian
Azimuthal angle of orb. ($t = 0$)	α_0	InitialAlphaAngle	Radian
Inclination of orbit	Λ	LambdaAngle	Radian
Luminosity distance	D_L	Distance	Gpc
time of plunge	t_{pl}	PlungeTime	Second
-	Approximant	AK	ModelName
-	ObservationDuration	-	Seconds
-	Cadence	-	Seconds

Table 2: Parametrization used for EMRI (AK).

One remark, the plunge time and ν_0 are related, so either of them could be used to parametrize the waveform. The command used to produce the data set:

```
$ python ProduceEMRI.py --verbose --seed=1234 \
 --param=Param_RadlerEMRI_AK.txt LDC1-2_EMRI_v1.hdf5
```

6.6.2 Augmented Analytic Kludge model

6.7 Galactic Binaries

6.7.1 The signal description in the source frame

The GW signal from Galactic white dwarf binaries in the source frame is defined as:

$$h_+^S = \mathcal{A}(1+\cos^2\iota)\cos(\Phi(t)) \tag{45}$$
$$h_\times^S = -2\mathcal{A}\cos\iota\sin(\Phi(t)) \tag{46}$$

The phase is described as

$$\Phi(t) = \phi_0 + 2\pi f_0 t + \pi \dot{f}_0 t^2, \tag{47}$$

where we have used the fact that the frequency evolves slowly and we retained only the first derivative of frequency. The subscript 0 implies that those are constants and given at the initial moment of time $t = 0$, and ϕ_0 is initial GW phase. The expression for the GW strain in the SSB frame is described in the section 6.1.

6.7.2 Implementation

The waveform generation is split into two parts: (i) generation of the response in the time domain and fourier transformed in the frequency domain (ii) adding the "fast" (the carrier signal) in the frequency domain. The main description of this method is given in [8]. The implementation is located in `software/Waveforms/Galaxy/`. The Radler data sets were produced using the following scripts `software/LDCpipeline/scripts/ProduceGB_Radler.py`, `software/LDCpipeline/scripts/Produce_VGB.py` The waveform generation was cross-validated agains the "LW" class and script which compares the waveforms is `software/LDCpipeline/scripts/GB_LW_FD_LC_SNR.py`. The command to produce the Galaxy is

```
$ python ProduceGB_Radler.py --verbose --seed=12345 \
--param=ParamsGalaxy.txt LDC1-4_GB_v1.hdf5
```

```
$ python ProduceGB_Radler.py --verbose --seed=12345 \
--param=ParamsVGB.txt LDC1-3_VGB_v1.hdf5
```

6.7.3 Parametrization

In the table 3 we summarize the parametrization of the GB signal.

Note that ϕ_0 is an initial GW phase.

6.7.4 Hierarchical triplets

6.7.5 Episodic accretion

6.8 Sollar mass binary BHs

We use the IMRPhenomD (the low frequency part of it) model to generate the signal in the frequency domain. We also apply the TDI response in the frequency domain [11]. We use the initial frequency (at $t = 0$) to parametrize the waveform and implement only the dominant harmonics $l = 2, m = \pm 2$. The duration of the signal in the frequency (time) domain is defined by the amount of evolution from the initial frequency, by the nyquist frequency and duration of the data set. Based on this many signal have abrupt cut-off in the frequency domain and the high frequency sources have cut also in time domain (due to Nyquist frequency).

Parameter	Notation	units
β	EclipticLatitude	Radian
λ	EclipticLongitude	Radian
$\mathcal{A}$	Amplitude	strain
f_0	Frequency	Hz
$\dot{f}_0$	FrequencyDerivative	Hz^2
ι	Inclination	Radian
ψ	Polarization	Radian
ϕ_0	InitialPhase	Radian
T_{obs}	ObservationDuration	Seconds
Δt	Cadence	Seconds

Table 3: Parametrization used for GB signal.

6.8.1 Implementation and parametrization

The parametrization of SOBBH signal is given in the table 4.

Parameter	Notation	units
β	EclipticLatitude	Radian
λ	EclipticLongitude	Radian
s_1	magnitude Spin1	MassSquared
s_2	magnitude Spin2	MassSquared
m_1	Mass1	SolarMass
m_2	Mass2	SolarMass
θ_{S1}	PolarAngleOfSpin1	Radian
θ_{S2}	PolarAngleOfSpin2	Radian
ϕ_{S1}	AzimuthalAngleOfSpin1	Radian
ϕ_{S2}	AzimuthalAngleOfSpin2	Radian
f_0	Frequency	Hz
ι	Inclination	Radian
ψ	Polarization	Radian
ϕ_0	InitialPhase	Radian
z	Redshift	dimensionless
D_L	Distance (luminosity)	Gpc
T_{obs}	ObservationDuration	Seconds
Δt	Cadence	Seconds

Table 4: Parametrization used for SOBBH signal.

Note again (like in the case of MBHB) the azimuthal angles of spins are redundant here (due to use of IMRPhenomD model). The redshift z and the luminocity distance are ralted by cosmological parameters (Planck2015). To be consistent with MBHB we have also use $\psi \rightarrow \psi_{FD} = \pi/2 - \psi$ and $\psi \rightarrow \psi_{LW} = \pi/2 + \psi$ modification for the polarization.

The waveform generation (including TDI) is implemented in `software/LDCpipeline/scripts/MakeSOBBH_FD.py` and the Raddler data production script is `software/LDCpipeline/scripts/ProduceSOBBHs.py` which was used to produce the data sets with full population and with only bright sources.

As before we have done comparison with "LW" class. Note that due to approximations done in "LW" and due to high frequency of SOBBH the very good agreement could be achieved only if the order of stencil is zero. The script which performs comparison and computes SNR of all bright sources is software/LDCpipeline/scripts/SOBBH_FD_LW_LC.py.

The commands to produce the SOBBH data sets are

```
$ python ProduceSOBBHs.py --verbose --seed=12345 \
--parFile=Param_BrightSOBBH.txt LDC1-5_SOBBH_Bright.hdf5
```

```
$ python ProduceSOBBHs.py --verbose --seed=12345 \
--parFile=Param_RadlerSOBBH.txt LDC1-5_SOBBH_FD.hdf5
```

6.9 Stochastic GW signal

7 Common tools

We have set of tools located in `software/Packages/common/` and they are installed as packages. We give a brief description of those modules here.

Cosmology This package computes luminosity/compving distance based on the redshift (and vice versa). We use Planck2015 parameters for the cosmological model.

GenerateFD SignalTDIs module used to generate the MBHB and SOBBH data with TDI response in frequency domain.

FrequencyArray module used to store the data in the frequency array. It comes with several useful features and is used for GB and SOBBHs.

tdi module to compute the noise TDIs (analytic) with or without Galactic confusion (fit) as well as sensitivity (approximate, analytic). And more...

LISAConstants this module is still the work in progress. The idea is to use the same constants for all codes (C, C++, python). Each constant should have a proper reference.

LISAhdf5 important module which is used to read and write the hdf5 files.

8 LISA Instrument

8.1 LISA orbits

8.1.1 Equal arm analytic orbit

This is very simple and not realistic orbits. We use it in the Radler data release. The orbital motion for each LISA spacecraft:

$$x_n = a\cos\alpha + ae\left(\sin\alpha\cos\alpha\sin\beta_n - (1+\sin^2\alpha)\cos\beta_n\right) \tag{48}$$

$$y_n = a\sin\alpha + ae\left(\sin\alpha\cos\alpha\cos\beta_n - (1+\cos^2\alpha)\sin\beta_n\right) \tag{49}$$

$$z_n = -\sqrt{3}ae\cos(\alpha-\beta_n) \tag{50}$$

where those are the coordinates in the solar system barycentric (SSB) frame and the phases are

$$\beta_n = (n-1)\frac{2\pi}{3} + \lambda \tag{51}$$

$$\alpha(t) = \frac{2\pi}{1\,\mathrm{year}}t + \kappa \tag{52}$$

and $\kappa = 0$, $\lambda = 0$ define the initial conditions. The parameter a is equal to 1 AU (Astronomical Unit). The orbital eccentricity is computed based on the armlength: $e = L/(2a\sqrt{3})$.

8.2 TDI noises

The intrumental noises are directly produced by the simulator.

The global noise levels are described in the file `software/Packages/common/LISAParameters.py` All noise sources are combined into two low and high frequency contributions:

- S^{Acc} : the PSD of the acceleration noise,
- S^{IMS} : the PSD of the Interferometric Measurement System.

8.3 TDI convention

Let us introduce the LISA constellation and notations, schematically it is given in the figure 4. The guiding center is moving around sun about 20° behind Earth and each spacecraft (labeled $1, 2, 4$) is in the cartwheeling motion around O. We have explicitly showing non-equal arm configuration, the separation between s/c is a function of time and is not equal pair-wise L_1, L_2, L_3. We have six independent measurement, two (back and forth) along each arm.

Introduce Michelson X-TDI variable , we work here with the first generation of TDIs:

$$X = y_{1-32,32-2} + y_{231,2-2} + y_{123,-2} + y_{3-21} - \\ [y_{123,-2-33} + y_{3-21,-33} + y_{1-32,3} + y_{231}] \tag{53}$$

The Y, Z channels are obtained by cyclic permutation of indices $1 \to 2 \to 3 \to 1$. We have used a short hand notation for delays:

$$y_{slr,i}(t) \equiv y_{slr}(t - L_i) \tag{54}$$

If the noise is uncorrelated and similar in each link the total noise in the X TDI (the same in Y, Z): The resulting noise PSD is

$$S_X = 16\sin^2\omega L\left[2(1+\cos^2\omega L)S^{Acc} + S^{IMS}\right]. \tag{55}$$

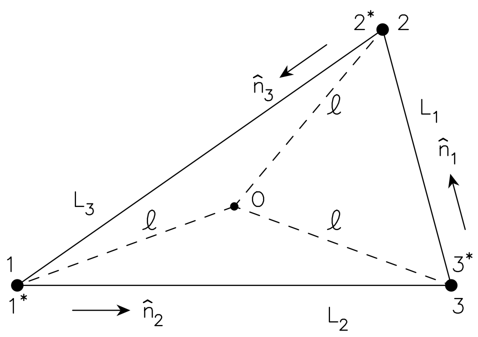

Figure 4: LISA's constellation.

Note that the noise is correlated between X, Y, Z with the cross-spectrum:

$$S_{XY} = -8\sin^2\omega L \cos\omega L \left(S^{IMS} + 4S^{Acc}\right) \tag{56}$$

We can construct the noise independent data combinations, called here A, E, T. Note that these combinations are not uniquely defined as

$$E = \frac{X - 2Y + Z}{\sqrt{6}}, \quad A = \frac{Z - X}{\sqrt{2}}, \quad T = \frac{X + Y + Z}{\sqrt{3}} \tag{57}$$

The numerical coefficients for A and E are chosen so that they have equal noise PSD:

$$\begin{aligned} S_A &= S_E = (S_X - S_{XY}) = \\ & 8\sin^2\omega L \left[4(1 + \cos\omega L + \cos^2\omega L)S^{Acc} + (2 + \cos\omega L)S^{IMS}\right] \end{aligned} \tag{58}$$

Similarly we can compute PSD for T: $S_T = S_X + 2S_{XY}$,

$$S_T = 32\sin^2\omega L \sin^2\frac{\omega L}{2}\left[4\sin^2\frac{\omega L}{2}S^{Acc} + S^{IMS}\right]. \tag{59}$$

Now let us describe the response to the GW signal. We start with the single arm response:

$$y^{GW}_{slr} = \frac{\Phi_l(t_s - \boldsymbol{k}\boldsymbol{R}_s(t_s)) - \Phi_l(t - \boldsymbol{k}\boldsymbol{R}_r(t))}{2(1 - \boldsymbol{k}\boldsymbol{n}_l)} \tag{60}$$

where the subscriots slr mean "s"ender, "l"ink, "r"eciever and

$$\Phi_l = \boldsymbol{n}_l h_{ij} \boldsymbol{n}_l \tag{61}$$

is the projection of the GW strain on the unit vector of link defined by a unit vector $\boldsymbol{n}_l$. The time t_s could be approximated as follows $t_s = t - |\boldsymbol{R}_r(t) - \boldsymbol{R}_s(t_s)| \approx t - L_l$, and $L\boldsymbol{n}_l \approx$

$\boldsymbol{R}_r(t) - \boldsymbol{R}_s(t_s)$. The vecors $\boldsymbol{R}_i$ define the position of "i"-th spacecraft. We assume here that arms are approximately equal $L_i \approx L$.

Using this approximation we have

$$y_{slr}^{GW} = \frac{\Phi_l(t - \boldsymbol{k}\boldsymbol{R}_s - L_l) - \Phi_l(t - \boldsymbol{k}\boldsymbol{R}_r)}{2(1 - \boldsymbol{k}\boldsymbol{n}_l)} \tag{62}$$

Substitute h_ij decomposed in the polarization basis in SSB we get

$$\Phi_l = (\boldsymbol{n}_l\boldsymbol{n}_l)^{ij} \left[h_+^S \mathcal{E}_{ij}^+ + h_\times^S \mathcal{E}_{ij}^\times \right] \equiv F_l^+ h_+^S + F_l^\times h_\times^S \tag{63}$$

where

$$\mathcal{E}_{ij}^+ = (-\epsilon_{ij}^+ \cos 2\psi + \epsilon_{ij}^\times \sin 2\psi) \tag{64}$$

$$\mathcal{E}_{ij}^\times = (-\epsilon_{ij}^+ \sin 2\psi - \epsilon_{ij}^\times \cos 2\psi) \tag{65}$$

and $h_{+,\times}^S$ are given in the source frame. It is convenient to write the GW signal in the complex form (if the signal is given in the Fourier domain we use it as is):

$$h_+ = A_+ \cos\Phi(t) \equiv \frac{1}{2} A_+ e^{i\Phi(t)} + c.c. \tag{66}$$

$$h_\times = A_\times \sin\Phi(t) \equiv -\frac{i}{2} A_\times e^{i\Phi(t)} + c.c. \tag{67}$$

We will drop the "complex conjugate" part assuming that it is always there. The further simplifications assume that the amplitudes is slowly varying functions of time, and that the orbital time scale is much smaller compared to 1 year (period of LISA's orbital motion). Then we can neglect all delays in the amplitude $A_{+,\times}$ and antenna functions F_l. We also separate the position of each s/c in SSB into position of the guiding center O and vectors connecting O and s/c:

$$\boldsymbol{R}_i = \boldsymbol{R}_o + \boldsymbol{q}_i$$

Note that $\boldsymbol{q}_s + \boldsymbol{n}_l L = \boldsymbol{q}_r$ and q_i are of order L. After a little bit of algebra we obtain

$$y_{slr}^{GW} = -i\frac{\omega L}{2} A_l e^{i\Phi(t - \boldsymbol{k}\boldsymbol{R}_0)} e^{-i\omega \boldsymbol{k}\boldsymbol{q}_r} \mathrm{sinc}\left[\frac{\omega L}{2}(1 - \boldsymbol{k}\boldsymbol{n}_l)\right] e^{-i\frac{\omega L}{2}(1 - \boldsymbol{k}\boldsymbol{n}_l)}, \tag{68}$$

where

$$A_l = \frac{1}{2}[A_+ F_+^l - i A_\times F_\times^l]. \tag{69}$$

In the long wavelength limit ($\omega L \ll 1$) and at leading order we get

$$y_{slr}^{LW} = -i\frac{\omega L}{2} A_l e^{i\Phi(t - \boldsymbol{k}\boldsymbol{R}_0)} \tag{70}$$

Now we compute the X-Michelson TDI given by expression (53). We again use approximation that the delays are small: $y_{slr,1} = y_{slr}(t - L_1) \approx y_{slr} e^{-i\omega L}$, and obtain

$$X = -2i \sin\omega L \left[(y_{1-32}^{GW} - y_{123}^{GW}) e^{-i\omega L} + y_{231}^{GW} - y_{3-21}^{GW}) \right] e^{-i\omega L} \tag{71}$$

We can also introduce the response R_X given as

$$X = R_X A e^{i\Phi(t - \boldsymbol{k}\boldsymbol{R}_0)}, \tag{72}$$

where the amplitude A is a common part of $A_+, A_\times$. We need to add complex conjugate to (71) (or to take twice the real part) to get the real expression. Other two Michelson combinations (Y, Z) are obtained by rotation of indices $1 \to 2 \to 3 \to 1$.

At the end we want to outline the the generation of the data in the frequency domain. The IMR waveform is often generated as set of modes (expansion in the spin-weighted spherical harmonics $Y_{lm}(\theta, \phi)$, we omit the spin weight index (-2)). For the non-precessing binaries, the dominant GW modes are:

$$\tilde{h}_+ = \frac{1}{2}(Y_{22} + Y^*_{2-2})A_{22}e^{i\Psi(f)} \tag{73}$$

$$\tilde{h}_\times = \frac{-i}{2}(Y_{22} - Y^*_{2-2})A_{22}e^{i\Psi(f)} \tag{74}$$

Assuming the convention for the Fourier transform as $\tilde{x}(f) = \int x(t)e^{-i2\pi ft}dt$, we get for the delays:

$$\Phi_l(t - \delta t) \to \tilde{\Phi}_l(f)e^{-i\omega\delta t}, \tag{75}$$

where with tilde we denote the functions in the Fourier (frequency) domain. Next we need to put the correspondence between time and instantaneous frequency of the GW mode:

$$t_f = -\frac{1}{2\pi}\frac{d\Psi_{lm}(f)}{df}. \tag{76}$$

In the first ("Radler") challenge we work only with the dominant modes. Note that the IMRPhenomD shifts the waveform so that the coalescence time corresponds to the $t = 0$, which needs to be corrected. The rest of derivations are similar to what we have done in the time domain. We refer to separate documentation which describes in details the generation of the response in the frequency domain beyond leading order [?]. The expression for the single link is the same as (68) with replacement

$$A_l e^{i\Phi(t-\boldsymbol{k}\boldsymbol{R}_0)} \to A_l e^{i(\Psi(f)-\omega\boldsymbol{k}\boldsymbol{R}_0)}.$$

9 For users

9.1 User request file

9.1.1 Common required parameters for the GW sources

- `SourceType` Type of source ; see 9.1.2
- `NumberSources` Number of sources of the given type

9.1.2 Source type (`SourceType`)

9.1.2.1 Massive Black Hole Binary (MBHB)

- `Catalogues` ...
- `CoalescenceTime`
- `MassRatio` ...
- `Spin1` ...
- `Spin2` ...
- `Model` ...
- `RequestSNR` ...
- `TimeStep` ...
- `ObservationDuration` ...

Example:

```
==============================

SourceType MBHB
NumberSources 1
Catalogues "catalogues/MBHs/catalog_Q3_delay_real106.out"
CoalescenceTime 0.1-0.25
MassRatio 1.0-10.0
Spin1 0.5-0.99
Spin2 0.5-0.99
Model IMRPhenomD
RequestSNR 100.0-500.0
TimeStep 10.0
ObservationDuration 1966065.0

==============================
```

9.1.2.2 extreme mass-ratio inspiral (EMRI) ...

9.1.2.3 Stochastic Gravitational Wave Background (SGWB)

- `Approximant` Approximative model use to generate the background. The only method implemented at the moment is `LISACode2SGWB`
- `Sky`: Sky distribution.
  - `Isotropic`
  - `Anisotropic` (not available at the moment)
- `FrequencyShape`: Frequency shape of the signal
  - `PowerLaw`: power law model defined as:

$$\Omega(f) = \Omega_0 \left(\frac{f}{f_*}\right)^\gamma \tag{77}$$

$$PSD(f) = \frac{3H_0^2\Omega(f)}{4\pi^2 f^3}\ Hz^{-1} = \frac{3H_0^2\Omega_0}{4\pi^2 f_*^\gamma} f^{\gamma-3}\ Hz^{-1} \tag{78}$$

    with $H_0 = 2.175 \times 10^{-18}$ Hz
  - `3PowerLaws` (Not yet implemented)
- `EnergyAmplitude`: Ω_0 Energy density or range of energy density. Only use for model `FrequencyShape` = `PowerLaw`.
- `EnergySlope`: γ Slope of the power law background or range of slope. Only use for `FrequencyShape` = `PowerLaw`)
- `FrequencyRef`: f_* reference frequncy in Hertz. Only use for model `FrequencyShape` = `PowerLaw`.

Example:

```
==============================

SourceType SGWB
NumberSources 1
Approximant LISACode2SGWB
Sky Isotropic
FrequencyShape PowerLaw
EnergySlope 0.666667
FrequencyRef 25
EnergyAmplitude 0.5e-9:4.5e-9

==============================
```

9.2 Catalogs

9.2.1 Massive Black Hole Binary (MBHB)

- MBHB catalogs are located in /catalogues/MBSs, and are those used in [10] (https://arxiv.org/pdf/1511.0 K16 hereinafter).
- There are 3 population models in 3 directories pop3, Q3_nodelays, Q3_delays. Description of the distinctive physical properties of each model can be found in K16.

- There are 10 files in each model directory. Each file contains *all merging MBHBs in the universe* in 1 yr (for a total of 10 yr of data).
- No selection based on LISA detectability has been performed to generate the catalogs.
- in each file, colums are:
  1. column 1: redshift of coalescence;
  2. column 2: intrinsic mass of primary (solar masses);
  3. column 3: intrinsic mass of secondary (solar masses);
  4. column 4: spin magnitude of primary;
  5. column 5: spin magnitude of secondary;
  6. column 6: angle between s_1 and L (L is the binary orbital angular momentum) in radians;
  7. column 7: angle between s_2 and L in radians;
  8. column 8: angle between the two projections of s_1 and s_2 into a plane perpendicular to L (i.e., in the plane of the binary) in radians. The angle is randomly picked in the range $[0, 2\pi]$;
  9. column 9: coalescence time in seconds (random between 0 and 5 years, this can be re-drawn...);
  10. column 10: θ sky location, uniform in $\cos\theta$;
  11. column 11: ϕ sky location, uniform in $[0, 2\pi]$;
  12. column 12: inclination ι defined as the angle between L and the line of sight, uniform in $\cos\iota$;

9.2.2 extreme mass-ratio inspiral (EMRI)

- EMRI catalogs are located in /catalogues/EMRIs, and are those used in [4] (https://arxiv.org/pdf/1703.09722.pdf, B17 hereinafter).
- There are 12 population models in 12 directories M1,...,M12. The labels correspond to the models reported in Table 1 of B17. Description of the distinctive physical properties of each model can be found in there.
- There are 10 files in each model directory. Each file contains EMRIs detected in 1 yr (for a total of 10 yr of data) with total SNR> 20 assuming the AKK waveform model. Note that in M7 and M12 every year catalog (EMRICAT101*, EMRICAT102*, etc) is subdivided in 3 or 4 subfiles denoted with SNR0, SNR1, etc (for computational purposes). The sum of each subfile is still worth 1 yr EMRI detections
- The content of the files are specified in the first row column headers; columns are:

... [TODO: Alberto/Enrico: describe the catalogs here]

9.3 Run the pipeline to generate data

We assume here you start from the docker image (see section 5.1).

9.3.1 Generate source list

- Start a configuration file. In the example, we copy `Ref_Param.txt` from `examples`:

```
cp /codes/LDC/examples/Ref_Param.txt Param.txt
```

- Choose the sources and produce the `hdf5`

```
ChooseSources.py --paramFile=Param.txt --filename=MySim_Param.hdf5 --seed=12345
```

- You can check the parameters using `LISADhdf5` `python3` class. Start `ipython3` then:

```
from LISAhdf5 import *
f = LISAhdf5('MySim_Param.hdf5')
print("Number of sources = ", f.getSourcesNum())
print("Name of sources = ", f.getSourcesName())
p = f.getSourceParameters('MBHB-0')
p.display()
```

9.3.2 Generate h_+ and $h_\times$

- Generate h_+ and $h_\times$:

```
Compute_hphc.py MySim_Param.hdf5
```

- To check the waveform. Start `ipython3` then:

```
from LISAhdf5 import *
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
LH = LISAhdf5('MySim_Param.hdf5')
thphc = LH.getSourceHpHc('MBHB-0')
plt.plot(thphc[:,0],thphc[:,1])
plt.xlabel('Time(s)')
plt.ylabel('h+')
plt.savefig('hphc_MBHB0.png')
```

9.3.3 Configure Instrument and Noises

- Configure instrument:

```
ConfigureInstrument.py --duration=31457265 MySim_Param.hdf5
```

- Duration is 1 year instead of 4 years for default. See `ConfigureInstrument.py --help` for more options (timestep, duration, TDI, orbits, ...). For example, if you want to use the MLDC orbits, add the option `--orbits=MLDC_Orbits`.
- Configure noises:

```
ConfigureNoises.py MySim_Param.hdf5
```

Options:

  – `--LevelRandom=`x : Specify randomisation of noises level in percent [default 0]

$$PSD_{level} = PSD_{level,0}\left(1+\frac{\alpha}{100}\right) \qquad \text{with random } \alpha \in [-x, x] \tag{79}$$

9.3.4 Run simulation

- Run simulation using LISACode2:

```
RunSimuLC2.py MySim_Param.hdf5
```

- To check the waveform. Start `ipython3` then:

```
from LISAhdf5 import *
import matplotlib
matplotlib.use('Agg')
import matplotlib.pyplot as plt
LH = LISAhdf5('MySim_Param.hdf5')
tTDI = LH.getPreProcessTDI()
plt.plot(tTDI[:,0],tTDI[:,1])
plt.xlabel('Time(s)')
plt.ylabel('TDI X')
plt.savefig('TDI_X.png')
```

9.4 Manipulate LISAhdf5 files

- General tool to manipulate `hdf5` file
  HDFView(https://support.hdfgroup.org/products/java/hdfview/)

- Display `LISAhdf5`

```
LISAh5_display.py MySim_Param.hdf5
```

- Edit `LISAhdf5`

```
LISAh5_edit.py MySim_Param.hdf5
```

- Extract TDI outputs from `LISAhdf5` in ASCII file:

```
LISAh5_tdi2ascii.py MySim_Param.hdf5 MySim_Param_TDI.txt
```

- Plot Power Spectral Density (PSD) of the TDI output from `LISAhdf5` (for more options see `psd.py -h`):

```
psd.py MySim_Param.hdf5
```

10 For developpers

11 Support and improvements

11.1 Issues

If you have any issues or problems have a look at list of Frequently Asked Questions (FAQ) below. If you don't find the solution, use the *Issues* section on gitlab (https://gitlab.in2p3.fr/stas/MLDC/issues) or send an email to LDC WG (lisa-ldc-helpdesk-l@in2p3.fr)

11.1.1 FAQ

Before asking/sending e-mail, please make sure you have read this manual.

References

[1] LSC algorithm library. http://www.lsc-group.phys.uwm.edu/lal.

[2] P. Ajith, M. Boyle, D. A. Brown, S. Fairhurst, M. Hannam, I. Hinder, S. Husa, B. Krishnan, R. A. Mercer, F. Ohme, C. D. Ott, J. S. Read, L. Santamaria, and J. T. Whelan. Data formats for numerical relativity waves. *ArXiv e-prints*, page arXiv:0709.0093, September 2007.

[3] K. A. Arnaud et al. An Overview of the second round of the Mock LISA Data Challenges. *Class. Quant. Grav.*, 24:S551–S564, 2007.

[4] S. Babak, J. Gair, A. Sesana, E. Barausse, C. F. Sopuerta, C. P. L. Berry, E. Berti, P. Amaro-Seoane, A. Petiteau, and A. Klein. Science with the space-based interferometer lisa. v. extreme mass-ratio inspirals. *Phys. Rev. D*, 95(10):103012, May 2017.

[5] Leor Barack and Curt Cutler. LISA capture sources: Approximate waveforms, signal-to-noise ratios, and parameter estimation accuracy. *Phys. Rev.*, D69:082005, 2004.

[6] Luc Blanchet. Gravitational Radiation from Post-Newtonian Sources and Inspiralling Compact Binaries. *Living Rev. Rel.*, 17:2, 2014.

[7] Alvin J. K. Chua, Christopher J. Moore, and Jonathan R. Gair. Augmented kludge waveforms for detecting extreme-mass-ratio inspirals. *Phys. Rev.*, D96(4):044005, 2017.

[8] Neil J. Cornish and Tyson B. Littenberg. Tests of Bayesian Model Selection Techniques for Gravitational Wave Astronomy. *Phys. Rev.*, D76:083006, 2007.

[9] Sebastian Khan, Sascha Husa, Mark Hannam, Frank Ohme, Michael Pürrer, Xisco Jiménez Forteza, and Alejandro Bohé. Frequency-domain gravitational waves from non-precessing black-hole binaries. II. A phenomenological model for the advanced detector era. *Phys. Rev.*, D93(4):044007, 2016.

[10] A. Klein, E. Barausse, A. Sesana, A. Petiteau, E. Berti, S. Babak, J. Gair, S. Aoudia, I. Hinder, F. Ohme, and B. Wardell. Science with the space-based interferometer eLISA: Supermassive black hole binaries. *Phys. Rev. D*, 93(2):024003, January 2016.

[11] Sylvain Marsat and John G. Baker. Fourier-domain modulations and delays of gravitational-wave signals. 2018.

[12] Harald Pfeiffer. Overview lal gravitational wave frame definitions. Technical Report LIGO-T1800226, LIGO Scientific Collaboration and Virgo Collaboration, 2018.

[13] P. Schmidt, I. W. Harry, and H. P. Pfeiffer. Numerical Relativity Injection Infrastructure. *ArXiv e-prints*, March 2017.