Explanation and Code Listing of the Gravitational Wave Detector Simulation

Abid Jeem

February 4, 2025

1 Overview

This document explains a Python code file that simulates the response of a network of gravitational wave detectors to an incoming gravitational wave signal. The code contains functions to:

- Transform tensors between different coordinate systems.
- Generate source/detector vectors from angular parameters.
- Compute change-of-basis matrices between the gravitational wave (GW) frame, the detector frame, and the Earth-centered (EC) frame.
- Construct the gravitational wave strain tensor and compute the detector response.
- Generate oscillatory (sine-Gaussian) signal terms, time arrays, and noise.
- Build model responses from randomized source angles and amplitudes.
- Simulate ("real") detector responses and compare them to the model responses.

All angles are expressed in radians, and all dimensional quantities are in SI units.

2 Code Explanation

2.1 Global Constants and Detector Parameters

The code begins by importing necessary libraries and defining key constants. These include:

- The number of detectors, gravitational wave polarizations, and modes.
- Detector-specific parameters such as location (latitude/declination, longitude/right ascension) and orientation.
- Physical constants like Earth's radius and the speed of light.

2.2 Tensor Transformation Functions

There are three functions to perform basis transformations on tensors:

- `transform_2_0_tensor`: Transforms a (2,0) tensor (with two contravariant indices) using the inverse of the change-of-basis matrix.
- `transform_1_1_tensor`: Transforms a mixed (1,1) tensor.

- `transform_0_2_tensor`: Transforms a (0,2) tensor (with two covariant indices) by directly applying the change-of-basis matrix.

Each function uses `np.einsum` to perform the necessary index contractions.

2.3 Source and Detector Vectors and Basis Functions

- `source_vector_from_angles` converts a list of three angles into a unit vector from the center of the Earth toward the source (or detector). Only the first two angles are needed.
- `change_basis_gw_to_ec` computes the change-of-basis matrix from the gravitational wave frame to the Earth-centered frame. It builds an orthonormal basis from the source vector, applies a polarization rotation, and then inverts the resulting matrix.
- `change_basis_detector_to_ec` computes the corresponding transformation for a detector using its location and orientation.

2.4 Detector Response and Beam Pattern Functions

- `detector_response` computes the scalar detector response (strain) by transforming both the detector response tensor and the gravitational wave strain tensor into the Earth-centered frame and contracting them.
- `beam_pattern_response_functions` returns the beam pattern functions F_+ and $F_\times$ of the detector. These are obtained by transforming the detector response tensor into the gravitational wave frame and then extracting the appropriate tensor components.

2.5 Time Delay and Oscillatory Terms Generation

- `time_delay_hanford_to_livingston` computes the time delay between the Hanford and Livingston detectors based on their positions and the direction of the gravitational wave.
- `generate_network_time_array` creates an array of time samples covering the signal's lifetime plus the maximum detector time delay.
- `generate_oscillatory_terms` generates sine-Gaussian oscillatory terms modulated by a Gaussian envelope. These terms simulate the gravitational wave signal at each time sample.

2.6 Model Generation Functions

- `generate_model_angles_array` creates randomized sets of source angles.
- `generate_model_amplitudes_array` generates randomized amplitude combinations for the gravitational wave signal.
- `generate_model_detector_responses` computes the expected detector responses (for both Hanford and Livingston) over time for each combination of model source angles and amplitudes.

2.7 Simulated (Real) Detector Responses with Noise

- `generate_noise_array` creates an array of noise values.
- `generate_real_detector_responses` simulates "real" detector responses by generating one set of true source angles and amplitudes, computing the responses for each detector, and adding noise.

2.8 Best-Fit Angle Comparison

The function `get_best_fit_angles_deltas` compares the simulated detector responses and source angles with those generated by the model. It computes several metrics:

1. The summed absolute difference between the real source angles and the closest model angles.
2. The summed absolute difference using the model angles that yield the best (minimum) detector response.
3. A weighted summed absolute difference based on an exponential weighting of the response differences.

3 Conclusion

This code provides a modular framework to simulate gravitational wave signals and detector responses. It includes routines for tensor basis transformations, signal generation, noise addition, and parameter estimation via model comparison. The detailed inline comments and docstrings make the code easier to understand and maintain.