AxiSEM3D inputs for use with Instaseis

In the GitHub in the `Simulations/` directory is a script called `make_simulation.py`, which sets up a simulation that will be compatible with Instaseis. If you want to set all the parameters yourself then how to do this is outlined below. There are five input files that are required for an AxiSEM3D simulation, and these should all be located in the `/input/` directory. Note that for some parameters there are differences between PZ and PX simulations.

File: `inparam.advanced.yaml`

This file contains numerous settings that are of little interest to the average user. There is only one parameter that can be set to optimise memory usage: `nproc_per_group` should be set to the number of processors per node for your HPC system.

File: `inparam.model.yaml`

This file contains various settings related to the models used by the simulation, both 1D and 3D.

- `exodus_mesh` must be set to the name of the `.e` file that you wish to use for the 1D model.
- `lat_lon_north_pole_mesh` should be set to `SOURCE`.
- `flattening_on_surface` should be set to `SPHERE` so that there is no ellipticity.
- There should be no 3D models for use with Instaseis, so `list_of_3D_models` should be set to `[]`.

File: `inparam.nr.yaml`

This file contains parameters governing the azimuthal resolution of the simulation and is an extremely important part of AxiSEM3D, however with 1D models and Instaseis this is trivial.

- `type_Nr` must be set to `CONSTANT`.
- `constant` should be set to `1` for PZ simulations are `3` for PX simulations. This parameter can be set higher than these values, but it is unnecessary and will be less efficient. Less than these values and the simulation won't be accurate.

File: `inparam.output.yaml`

This file contains settings about how the information about the wavefield is to be outputted. Note that at present, the conversion code can only handle a maximum of one element group.

- There is no need to have any stations so `list_of_station_groups` can be set to `[]`. There is no harm in having stations as well, they are just unnecessary in most cases.
- `horizontal_range` controls the extent of the database in the distance direction in radians. The most general setting is `[0, 3.1415926535897]`, which will calculate the database to a distance of 180° from the source. To only calculate to 90°, for example, one could set this to `[0, 1.57079632679]`.
- `vertical_range` controls the extent of the database in the radial direction in kilometres, and therefore the range of source depths for which seismograms can be calculated. `[5671000, 6371000]`, which will allow sources as deep as 700 km, is a sensible choice. To reduce the maximum source depth (and save a lot of memory) then increase the first number.
- `edge_dimension` must be set to `BOTH`.
- `GLL_points_one_edge` should be set to `[0,1,2,3,4]` in order to record all GLL points, in which case Instasuse uses polynomial order $n = 4$. Set to `[0,2,4]` to only record nodes and edge centres, in which case Instaseis uses $n = 2$.
- `phi_list` and `lat_lon_list` should be `[]`.
- `na_space` should be `1`.
- `coordinate_frame` must be set to `spz`.
- `medium` should be `SOLID`, as the mantle is, well, solid.
- `channels` is required to contain `U`, any other entries are unnecessary.
- `sampling_period` can be set to `DTx40`. It can be smaller but then the databases will be larger. It should probably not be larger.
- `time_window` must be `FULL`.

File: `inparam.source.yaml`

This file contains all of the settings related to the source. Note that there can only be one source.

- `record_length` should be however long in seconds that you want your seismograms to be.
- `enforced_dt` should be `NONE` to use the recommended time step for the mesh.
- `latitude_longitude` should be set to `[90, 0]` so that the source is located at the north pole.

- `depth` should be set to the depth that your receivers will be (most usually 0 as receivers are most often at the surface).
- `type` should be set to `FORCE_VECTOR`.
- `data` should be set to `[1, 0, 0]` for a PZ source (AxiSEM: vertforce) and to `[0, 1, 0]` for a PX source (AxiSEM: thetaforce).
- `unit` then sets the magnitude of the source. `1e20` is consistent with original AxiSEM, but it doesn't matter too much.
- `class_name` must be `GaussianSTF`.
- `half_duration` should be set to `0`.
- `decay_factor` controls the shape of the STF, a value of 3.5 is consistent with AxiSEM.
- `time_shift` should be `0`.
- `use_derivative_integral` can be set to whichever of the options you prefer. `GAUSSIAN` or `ERF` would be the usual choices.