EDtoolbox - Input data structs

geoinputdata

Five alternatives exist for specifying the struct `geoinputdata`:
1. If the field `.freefieldcase` is given the value 1, then no scattering object will be defined and all other fields are irrelevant.
2. An external `.cad`-file can be specified in the field `.geoinputfile`.
3. If the field `.geoinputfile` is not specified, then the fields `.corners` and `.planecorners` should give the geometry data (but see alt. 4).
4. If neither of the two alternatives above apply (e.g., if the entire struct is left empty), then a file opening window will appear, and a `.cad`-file can be selected.
5. If both `.geoinputfile`, and `.corners`+`.planecorners` are specified, priority will be given to the `.geoinputfile`.

Field name	Description
`.geoinputfile`	(Optional) This field should be given the file name (with path) of a .cad-file
`.corners`	(Optional) A matrix of size [ncorners,3], giving the x-, y-, and z-coordinates of each corner.
`.planecorners`	(Optional) A matrix of size [nplanes,nmaxcornersperplane]. For each plane, the corners should be given in the counterclockwise direction when looking at the plane from the active side. Each row might need extra zeros to give all rows the same number of elements.
`.planerefltypes`	(Optional) Allowed values are 1 (default = rigid), 0 (ideal total absorption), and -1 (soft/pressure-release).
`.listofedgestoskip`	(Optional) Some edges can be deactivated by using this parameter, which can have one or two columns. Single-column ⇒ list of edge numbers will be deactivated. Two columns ⇒ each row has a corner number pair, the edge between which will be deactivated. Takes precedence over `.firstcornertoskip` and `.listofcornerstoskip`. Default value [] (empty).
`.firstcornertoskip`	(Optional) All edges with at least one corner with a number same or larger than `.firstcornertoskip` will be deactivated. Default value 1e6.
`.listofcornerstoskip`	(Optional) All edges with at least one corner belonging to the list `.listofcornerstoskip` will be deactivated. Default value [] (empty).
`.planeseesplanestrategy`	(Optional) This parameter is irrelevant for convex-shaped scattering objects but is preparing for non-convex objects. If this parameter is given the value 1, then a plane-to-plane visibility check is done by checking plane-midpoint to plane-midpoint visibility. Default value 0.
`.freefieldcase`	(Optional) If this parameter is given the value 1, then a free-field case will be run and no scattering object will be defined. Default value 0.

envdata

Field name	Description
`.cair`	(Optional) Should be the speed of sound in m/s. Default value 344.
`.rhoair`	(Optional) Should be the density of the medium, in kg/m^3. Default value 1.21.

controlparameters

Field name	Description
Maximum one of the three following parameters can be set to one:	*Chooses which calculation method should be used. If none is set to one, then only the geometry part will be run: identify the edges of the geometry model.*
`.docalctf`	Determines if transfer functions will be computed, with the ESIE method for higher-order diffraction.
`.docalctf_ESIEBEM`	Determines if transfer functions will be computed, with the ESIEBEM method.
`.docalcir`	Determines if impulse responses will be computed, with the "separate diffraction orders" method.
`.frequencies`	(Required if `.docalctf = 1` or `.docalctf_ESIEBEM`=1). A vector of frequency values that computations will be made for.
`.directsound`	(Optional) If this parameter is set to 0, the direct sound will not be computed and the result variable `tfdirect` or `irdirect` will be empty. Default value 1.
`.skipfirstorder`	(Optional) If this parameter is set to 1, the direct sound, specular reflection, and first-order diffraction will not be computed and their respective output variables will be empty. Default value 0.
`.Rstart`	(Optional) Determines the phase of the final transfer functions (or the definition of the time zero in TD calculations).[1] Default value 0.
`.difforder`	(Optional) Specifies how many orders of diffraction should be included.[2] Default value 10.
`.discretizationtype`	Determines how the edges will be discretized: 0 ⇒ a uniform discretization. 2 ⇒ Gauss-Legendre discretization. The value 1 is obsolete/not used. Default value 2.
`.ngauss`	Specifies the number of quadrature points along the longest edge, for the ESIE and ESIEBEM methods. It will be scaled down linearly based on the length of each edge, with a minimum of 2. Recommended value is at least 3 quadrature points per wavelength, which has to be converted manually to a number of quadrature points. Default value 16.
`.surfacegaussorder`	For the ESIEBEM method, this parameter determines how many intermediate receivers will be placed across each plane. A value of 5 implies 5*5 surface receivers for the largest plane, and progressively fewer for smaller planes. Default value 5.
`.fs`	Determines the sampling frequency. Default value 44100.
`.savealldifforders`	If this parameter is set to 1, then results are stored for each diffraction order. Default value 0.

[1] To simulate an incoming plane wave with amplitude 1, and phase zero, at the origin, then `.Rstart` should be set to the distance to the far-away monopole source. See also the description of the input data struct `Sinputdata`.
[2] For time-domain (impulse response) calculations, `difforder` can not be higher than 6.

Sinputdata

Field name	Description
.coordinates	(Required) A matrix of size [nsources,3], giving the x-, y-, and z- coordinates of each source.[1]
.doaddsources	(Optional) If this value is set to 1, the contributions from all sources will be added and saved in a single transfer function or impulse response, after being multiplied by the values in the matrix .sourceamplitudes. Default value 0.
.sourceamplitudes	(Optional) A matrix of amplitudes, size [nsources,nfreq], that each source is multiplied with. Using this, together with .doaddsources=1, a vibration pattern on a surface can be simulated.[2] Default value ones(nsources,nfreq).
.doallSRcombinations	(Optional) If n sources and n receivers are specified, one can choose to compute the response only for source 1 to receiver 1, source 2 to receiver 2, etc by setting this parameter to 0. This is relevant for computing monostatic backscattering. Default value 1.
.sourcetype	(Optional) Can be either monopole (default) or polygonpiston.
.pistoncornercoordinates	(Optional) but (Required) if .sourcetype is polygonpiston. Format as for geoinputdata.corners, that is, x, y, z of all npistoncorners piston corners, stored in a matrix of size [npistoncorners,3]. Please note that these coordinates must be given values that fulfill the plane equation of the plane specified in the field .pistonplanes. The EDtoolbox does not (as of version 0.501) check this for you.
.pistoncornernumbers	(Optional) but (Required) if .sourcetype is polygonpiston. Matrix of size [npistons,maxncornersperpiston], where each row contains the 3,4,5,.... corner numbers for each piston. The order must follow the same right-hand rule as the planecorners: if you put your right hand to point in the direction of the plane and piston normal vector, then your curved fingers should point in the direction of the corner order. This can also be expressed as counterclockwise order when one is looking at the piston from the outside of the scattering object. Please note that if the pistons have different numbers of corners, the matrix in .pistoncornernumbers must get zeros at the end of each row which has fewer than maxncornersperpiston values.
.pistonplanes	(Optional) but (Required) if .sourcetype is polygonpiston. List of size [npistons,1], which gives the plane number (defined in the list geoinputdata.planecorners) that each piston belongs to.
.pistongaussorder	(Optional); only relevant if .sourcetype is polygonpiston. A number which will have slightly different meanings for different piston types. The default value is 3. · For a rectangular piston: the number of gauss quadrature points along one dimension of each piston. The total number of gauss quadrature points is thus .pistongaussorder^2. · For a triangular piston: .pistongaussorder with values [1, 2, 3, 4, 5, 6, 7, 8] gives, respectively, [1, 3, 4, 6, 7, 12, 16, 16] quadrature points. · For a regular polygon (approximating a circle), quadrature points will be distributed in .pistongaussorder concentric circles with linearly distributed radii. For each radius, the number of quadrature points is [1, 8, 16, 24, 32, ...] so that for .pistongaussorder = [1, 2, 3, 4, 5, ...], the total number of points is, respectively, [1, 9, 25,49, 81, ...].

[1] If a source is placed at a surface, it needs to be placed a tiny distance away from the surface, say 10^{-5} m.

[2] To simulate an incoming plane wave with amplitude 1 at the origin, then .sourceamplitudes should be given the value "distance", where "distance" is the distance to the far-away monopole source. See also the description for the input data struct controlparameters. It is also possible to implement this by scaling the resulting transfer functions/impulse response accordingly.

Rinputdata

Field name	Description
.coordinates	(Required) A matrix of size [nreceivers,3], giving the x-, y-, and z- coordinates of each receiver.[1]

[1] If a receiver is placed at a surface, it needs to be placed a tiny distance away from the surface, say 10^{-5} m.

Filehandlingparameters

Field name	Description
.outputdirectory	(Optional) but (Required) if geometry is defined as .corners and planecorners. All result files will be saved in this directory. If not specified, a folder named "results" will be made in the folder of the cad-file.
.filestem	(Optional) but (Required) if geometry is defined as .corners and planecorners. All result files will start with this text string. If not specified, this field will be given the name of the cad-file.
.suppressresultrecycling	(Optional) 0 ⇒ all result files will be inspected for possible recycling. Default value 0.
.savecadgeofile	(Optional) 1 ⇒ the contents of the .cad-file will be saved in a _cadgeo.mat file. Default value 0.
.saveSRdatafiles	(Optional) 1 ⇒ the visibility of planes and edges, as seen from sources and receivers, is stored in _Sdata.mat and _Rdata.mat files. Default value 1.
.saveeddatafile	(Optional) 1 ⇒ the edgedata struct is saved in an _eddata.mat file. Default value 1.
.saveed2datafile	(Optional) 1 ⇒ the edgetoedgedata struct is saved in an _ed2data.mat file. Default value 1.
.savesubmatrixdata	(Optional) 1 ⇒ the submatrixdata struct is saved in a _submatrixdata.mat file. Default value 1.
.saveinteqsousigs	(Optional) 1 ⇒ the edge source signals are saved in a _sousigs.mat file. Default value 0.
.loadinteqsousigs	(Optional) 1 ⇒ previously calculated, and saved, edge source signals are loaded and reused. Default value 0.
.savepathsfile	(Optional) 1 ⇒ the lists of possible direct sound, specular reflections, and first-order diffractions, are saved in a _paths.mat file. Default value 1.
.savehodpaths	(Optional) 1 ⇒ the lists of possible higer-order diffractions are saved in a _hodpaths.mat file. Used only by EDmain_convex_time. Default value 0.
.savelogfile	(Optional) 1 ⇒ a log text-file is saved, see Section ??. Default value 1.
.savediff2result	(Optional) 1 ⇒ the results for second-order diffraction are saved separately, in the form of the variable extraoutputdata.tfinteqdiff_nodiff2. Default value 0.
.showtext	(Optional) 1 ⇒ some progression text is printed out on screen. If the value is set to 0, no text is printed out on the screen. If values higher than 1 are set, then even more detailed information is printed out on screen.