NIFITS 0.4 cheatsheet

The NIFITS team

August 2025

- `NIFITS` standard version 0.4
- `nifits` library version 0.0.6

List of Tables

1 The array information

The OIFITS format provides, in addition to the `OI_ARRAY` data, the baseline coordinates for each row of the recorded visibility information, which inform the user on the projected array. In NIFITS, this cannot be done by baseline, but the format should still alleviate the burden of computing the effective array layout. The coordinates (X, Y) of of each subaperture of the array, projected into the plane orthogonal to the line of sight, and aligned in rotation to the sky coordinates (α, δ) are included in `NI_MOD` for each frame. They should be given in units of meters (m).

The `OI_WAVELENGTH` is also kept from the OIFITS standard so as to facilitate insertion into existing databases.

Table 1: Summary of the **NIFITS extensions**

Extension	Required	Content
`OI_ARRAY`	yes	Interferometer description for compatibility with OIFITS.
`OI_WAVELENGTH`	yes	Contains the information about wavelength bins.
`NI_OSWAVELENGTH`	no	Contains the information about oversampled wavenlength bins.
`NI_DSAMP`	no	Contains a matrix allowing a downsampling to the wavenlengths of simulation (for `NI_OSWAVELENGTH` to `OI_WAVELENGTH`). Identity assumed if absent.
`NI_MOD`	yes	Contains the time-varying information of the model, in particular the an internal modulation phasor vector, and the projected location of collecting apertures.
`NI_CATM`	referenced	The complex amplitude transfer matrix containing all static behavior of the system.
`NI_KMAT`	no	Kernel combination matrix for the linear combination of outputs. Identity is assumed if absent.
`NI_IOUT`	yes	Contains the collected output flux.
`NI_KIOUT`	no	Contains post-processed output fluxes.
`NI_FOV`	referenced	Contains the complex spatial filtering function.
`NI_KCOV`	no	Contains covariance matrices corresponding to NI_KIOUT.

Table 2: `NI_MOD`: The table of time-dependent collectorwise information.

Item	format	unit	comment
`APP_INDEX`	int	NA	Index of subaperture (starts at 0)
`TARGET_ID`	int	d	Index of target in `OI_TARGET`
`TIME`	float	s	Backwards compatibility
`MJD`	float	day	
`INT_TIME`	float	s	Exposure time
`MOD_PHAS`	n_λ complex	NA	Complex phasor of modulation for the collector
`APPXY`	2 float	m	Projected location of subapertures in the plane orthogonal to the line of sight and oriented as (α, δ)
`ARRCOL`	float	m^2	Collecting area of the subaperture
`FOV_INDEX`	int	NA	The entry of the `NI_FOV` to use for this subaperture.

Table 3: `NI_CATM`: The complex amplitude transfer matrix. $n_{apertures}$ is the number of collecting apertures of the array, and $n_{outputs}$ is the number of outputs of the beam-combiner.

Item	format	unit	comment
`M`	$2 \times n_\lambda \times n_{outputs} \times n_{apertures}$ float	NA	The complex amplitude transfer matrix given for all wavelengths. Due to the limitation of the FITS standard, it is stored as real-values with the first dimension containing real and imaginary parts.

Table 4: `NI_KMAT`: the post-processing matrix, or kernel matrix (optional).

Item	format	unit	comment
`u`	$n_{outputs} \times n_k$ float	NA	Matrix representing the linear combinations of outputs to extract the relevant observables to be used in the inference.

Table 5: `NI_IOUT`: the output flux or counts.

Keyword	format	comment
`IUNIT`	string	The unit a used for output flux. The format must be ingestible by `astropy.units.Unit()`. e.g. `ph / s` .

Item	format	unit	comment
`u`	$n_\lambda \times n_{outputs}$ float	a	The data recorded during the observation either in unit a of photon flux or count flux, or other relevant unit specified in the header.

Table 6: `NI_KIOUT`: the post-processed outputs (optional).

Keyword	format	comment
`IUNIT`	string	The unit a used for output flux. The format must be ingestable by `astropy.units.Unit()`. e.g. `ph / s` .

Item	format	unit	comment
`u`	$n_\lambda \times n_k$ float	a	Post-processed outputs recorded during the observation.

Table 7: NI_KCOV: the covariance of post-processed outputs (optional).

Keyword	format	comment
IUNIT	string	The unit used for output flux. The format must be ingestable by `astropy.units.Unit()`. e.g. `ph2 / s2` .

Item	format	unit	comment
u	$n_\lambda \times n_k$ float	a^2	An estimate of the covariance matrix reflecting the error on KIOUT data.

2 Field of view

Spatial filtering is an important element of the many nulling instruments. This effect is a phasor represented by a complex-valued function of wavelength and at least one spatial dimension projected on sky. The spectral dimension is discretized onto the n_λ spectral channels, but the spatial dimension must allow for a continuous representation. The nature of this function is defined by the keyword `FOV_MODE` of the `NI_FOV` extension. This first version of NIFITS implements a single function for the keyword `diameter_gaussian_radial` which is a simple wavelength-dependent gaussian fiber mode, including a wavelength-dependent offset. The offset is stored in the `offsets` column of the table for each recorded frame. While straightforward to implement for default values, its offset allows for the inclusion of transverse dispersion effects incurred by ground-based observatories such as the coming Asgard/NOTT and make it a powerful tool.

The injection optics of spatial filters can lead to phase and amplitude effects over the field of view that can bias the measurements (seen e.g. for GRAVITY). For this reason, the standard should allow the provision for powerful tools to model its effect on the signal.

Future upgrades to this format may include ways to facilitate chromatic shift and rotation of these functions, so as to implement atmospheric dispersion effects without repeating of the `NI_FOV` matrix.

Table 8: NI_FOV: the spatial filtering phasor function.

Keyword	format	comment
FOV_MODE	string	Corresponds to a type of function to represent the injection e.g. diameter_gaussian_radial.
FOV_TELDIAM	float	The collecting diameter.
FOV_TELDIAM_UNIT	string	A string giving the unit. Must be readable by astropy.units .

Item	format	unit	comment
INDEX	int	NA	The index of the row.
offsets	$(2\times)n_x$ float	mas	Offset of the center of the mode with respect to the center of the field of view
q	$(2\times)n_{lambda} \times n_x$ complex	NA	The phasor values at the sampled points.

3 Basic guidelines

The standard is built to have a single interpretation for the user. However, for the creator of the file, some redundancy in the various extensions open choices to the creator in the way their data is stored. It is of the responsibility of the creator to ensure the consistency of the file and avoid duplication of effects. For example, a known optical path residual could be stored in the NI_MOD aray, or factored into the NI_CATM array. For these cases, the guidelines to observe are :

1. The amplitude effects must be given only once either :
   - Through the modulus of the NI_MOD array for time-varying effects.
   - Through the moduli of the NI_CATM array for fixed effects.
   - Through the collecting aperture of the collectors in OI_ARRAY,
   - Through the field of view function (not recommended).
2. The phase effects should be given only once either :
   - Through the argument of the NI_MOD array for time varying effects.
   - Through the arguments of the NI_CATM array for fixed effects.

To put it simply:

- Effects considered static over a number of observations and possibly several nights should be factored into NI_CATM, so it can be shared by different files. This applies in particular to effects that arise inside the beam-combiner and can be measured with a calibration source.
- Effects that are variable should be factored into NI_MOD. This includes intentional and unintentional effects e.g.:

Table 9: `NI_DSAMP`: the spectral downsampling matrix (optional).

Item	format	unit	comment
`u`	$n_\lambda \times n_{\lambda u}$ float	NA	The spectral down-sampling matrix.

Table 10: `NI_OSWAVELENGTH`: the table of oversampled wavelength.

Item	format	unit	comment
`EFF_WAVE`	n_λ float	m	The central wavelength of each spectral bin.
`EFF_BAND`	n_λ float	m	The bandwidth of each spectral bin.

- The π phase shift to obtain a null,
- Active modulation of the phase or amplitude,
- Known or expected residual from atmospheric dispersion,
- Measured phase error from metrology,
- Known effects of the local weather conditions and status of the collectors,

4 Spectral channel oversampling

For low spectral resolution and broadband instruments it may be important to model the expected behavior of the system at spectral resolution higher than that offered by the instrument itself. NIFITS offers this as optional capability. In this case, the number of spectral channels used throughout the file corresponds to an oversampled number, while the `OI_WAVELENGTH` definition remains that of the measurement matching the size of the output vector $\boldsymbol{g}_o$, for consistency of metadata referencing, but an additional table `NI_OSWAVELENGTH` of similar format contains the information of the oversampled bins.

An additional matrix is then provided to inform of the linear combination that constitute the reduction, typically with rows containing series of consecutive ones indicating the values that are coadded, but it may contain more complicated convolution kernels of your spectrograph. This matrix can be stored in `NI_DSAMP` containing a matrix $\boldsymbol{N}$ which multiplies our model operation on the left for each output o:

$$\boldsymbol{g}_o = \boldsymbol{N} \cdot \Big[\boldsymbol{g}_\lambda\Big]. \quad (1)$$