PROCEEDINGS OF SCIENCE

NuRadioReco: A new reconstruction framework for radio neutrino detectors

Christian Glaser*[a], Anna Nelles[b,c], Ilse Plaisier[b,c], Christoph Welling[b,c], Steven W. Barwick[a], Daniel García-Fernández[b,c],Geoffrey Gaswint[a], Robert Lahmann[a,b], Christopher Persichilli[a]

[a] *University of California, Irvine, California, USA*
b ECAP, Friedrich-Alexander-Universität Erlangen-Nürnberg, Erlangen, Germany
c DESY, Platanenallee 6, 15738, Zeuthen, Germany
E-mail: christian.glaser@uci.edu

The radio detection of high-energy neutrinos has huge potential as it allows for cost-efficient instrumentation of huge volumes leading to unprecedented sensitivity in the 10^{16} eV - 10^{20} eV energy regime. The required technological advances of the detector hardware have matured in the two pilot arrays ARA and ARIANNA and the focus shifts to analysis and reconstruction techniques. We present a new modular framework for the detector simulation and data reconstruction of radio detectors of neutrinos and cosmic rays. The software is written in Python and publicly available on github. We built on the long-standing experience of cosmic-ray detectors as many aspects of data processing, detector simulation and reconstruction are very similar for neutrinos. NuRadioReco is already being used by the ARIANNA experiment and for the detector simulation in the novel neutrino MC code NuRadioMC. It was designed to be flexible so that is can be adapted for different neutrino and cosmic-ray detectors and can serve as the software framework of a future large-scale Askaryan detector which is actively discussed in the community.

36th International Cosmic Ray Conference -ICRC2019-
July 24th - August 1st, 2019
Madison, WI, U.S.A.

*Speaker.

1. Introduction

NuRadioReco is a modular Python framework for the detector simulation and event reconstruction of radio detectors for cosmic rays and neutrinos. Its design built up on long standing experience with the software needs of large radio cosmic-ray experiments [1, 2, 3, 4] as well as with Monte Carlo studies and data analysis of cosmic-ray and neutrino detectors [5, 6, 7, 8, 9, 10]. This allowed to combine the best of previous software while addressing some of its shortcomings. The main design criterion was user friendliness and ease of use. This keeps the entry barrier low for incoming students and allows a productive use of the software without a steep learning curve.

All code is open source and the software is developed collaboratively on github [11] making full use of the provided features such as code review, issue reporting, milestones and continuous integration testing. This allows for an efficient development, coordination and testing across multiple research groups working on the software.

NuRadioReco follows a clear modular design with a strict differentiation between detector description, event data and the processing steps. Each analysis step, or step in the detector simulation such as unfolding of the amplifier response, is encapsulated in its own processing module. This allows for a simultaneous and independent development of modules, and easy exchange and testing of different reconstruction strategies.

NuRadioReco not only implements all standard processing steps of a radio detector but also comes with a novel algorithm to reconstruct the electric field which substantially improves the resolution of the polarization and frequency spectrum. A comprehensive description of NuRadioReco and its novel reconstruction techniques is provided in [12]. In the following we give a brief summary of the main features of NuRadioReco.

2. Default system of units

Most variables in physics analyses carry a unit and not correctly keeping track of the used units is a common mistake. To prevent this, NuRadioReco employs a default system of units, a concept adapted from [1]. Every time a variable with unit is defined, it is multiplied with the corresponding unit. Then, all calculations done internally can be performed without considering units as all variables will be in the same default system of units. Only if the user wants to print or plot a variable in a certain unit, it needs to be divided by the unit. This is illustrated in the following snippet.

```python
from NuRadioReco.utilities import units
time = 132. * units.ms # define 132 milli seconds
d = 5. * units.mm # define 5 mm
v = d/time  # calculate speed
print("the speed is {:.2f} km/h".format(v/units.km*units.hour))
# the speed is 0.14 km/h
```

3. Parameter storage

NuRadioReco implements an easy and well defined way to store parameters into the event structure adapted from [2]. A developer only needs to define the parameter to an `enum`. Then, the parameter can be set/retrieved via a generic *set/get_parameter* function, or through a dictionary-like interface. This substantially reduces code compared to the traditional way of defining new member variables, writing a getter/setter function, and adding the variable to the output file. The usage is demonstrated in the following snippet:

```
from NuRadioReco.framework.parameters import stationParameters as stnp
from NuRadioReco.utilities import units

# set parameters via generic setter function
station.set_parameter(stnp.nu_energy, 7e8 * units.GeV)
# or via dictionary like interface
station[stnp.nu_energy] = 7e8 * units.GeV
# set uncertainty of neutrino energy
station.set_parameter_error(stnp.nu_energy, 1e6 * units.GeV)
# access of parameters
nu_energy = station.get_parameter(stnp.nu_energy)
# or
nu_energy = station[stnp.nu_energy]
```

4. Input/output

All event data is stored in a hierarchical class structure in memory. For example, an event has stations which have channels that contain the voltage traces. Each class provides its own (de)serialization function, a concept adapted from [4]. Thus, an event can be saved to disk by calling the serialization function of the top-level event class which will recursively call the serialization of its children. In the same way, files are read in: Calling the deserialization function of the event class on a NuRadioReco file will restore the full hierarchical class structure back into memory.

This has the huge advantage that the input file format is the same as the output file format. Hence, a detector simulation or reconstruction can be subdivided into several steps where each intermediate result is saved to disk. This way, time consuming low level processing steps have to be calculated only once. This intermediate result can then be processed several times with different high-level reconstruction algorithms.

All parameters on station and event level are stored in an event header which allows for a quick reading and plotting of high-level quantities. With the following few lines of code, a histogram of the reconstructed zenith and azimuth direction is plotted.

```
import NuRadioReco.modules.io.NuRadioRecoio as NuRadioRecoio
from NuRadioReco.utilities import units
from NuRadioReco.framework.parameters import stationParameters as stnp
import matplotlib.pyplot as plt

nurio = NuRadioRecoio.NuRadioRecoio("my_file.nur")
header = nurio.get_header()
station_id = 51
station_header = header[station_id]

# get numpy arrays of reconstructed direction
zeniths_rec = station_header[stnp.zenith]
```

```
azimuths_rec = station_header[stnp.azimuth]

# plot zenith vs azimuth in degrees
plt.plot(zeniths_rec/units.deg, azimuths_rec/units.deg)
plt.show()
```

Apart from the dedicated NuRadioReco file format, input modules for other sources are provided as well. E.g., a reader for CoREAS air shower simulations as well as for ARIANNA and ARA data is provided.

5. Detector description

NuRadioReco comes with a SQL data base design that allows to save all relevant detector information efficiently and is already designed for the needs of a large-scale experiment. Although such a design is required for a large detector, using SQL data bases come with disadvantages for the user. Therefore, the detector description can also be defined in a human-readable JSON file, and a converter from SQL to JSON is provided. Thus, when running NuRadioReco, no remote connection to an SQL server is required. Especially for multi-processing on a cluster this is a big advantage as many open SQL connections are a typical bottleneck. This design also allows for an easy set up of detector descriptions for simulation studies.

A critical part in the simulation and reconstruction of radio detector data is the antenna response. NuRadioReco provides a convenient interface to simulations of the antenna response, and interfaces to typically used programs such as WIPL-D, XFDTD and NEC2. The antenna response class handles all coordinate rotations that allows to orient the antenna into any direction and interpolates the simulated responses to arbitrary directions and frequencies. NuRadioReco also comprises a variety of antenna models used in the radio detection of neutrinos, such as LPDA and bicone antennas.

6. Event visualization

NuRadioReco provides an event browser that uses state-of-the-art web technologies and builds up on the dash framework. This allows for a platform-independent implementation as users can use their favorite web browser to inspect the data files. A local web server to inspect data locally can be started via the command

```
NuRadioViewer /path/to/files
```

which will directly open the event viewer in the default web browser. Thus, it can be used like a traditional GUI application.

Alternatively a remote web server can be set up which opens up new possibilities for outreach activities and collaborative sharing. Data can even be inspected from a tablet or smartphone, the only software requirement being a web browser. A screenshot of the eventbrowser is presented in Fig. 1.

PoS(ICRC2019)900

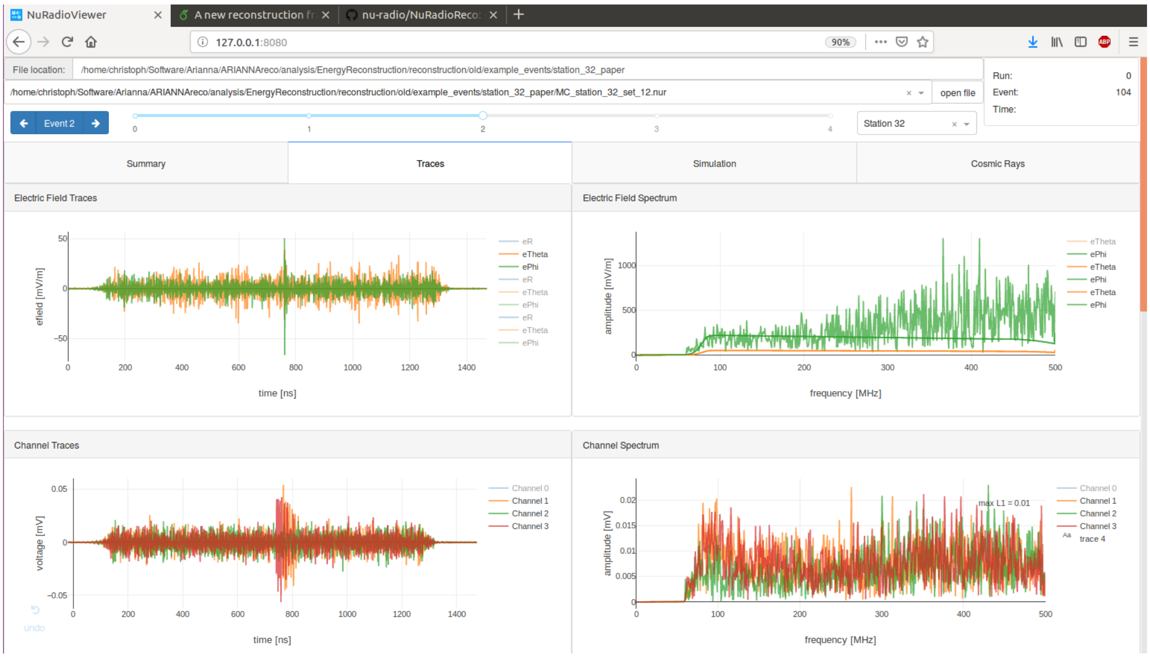

Figure 1: Screenshot of the EventBrowser. The EventBrowser can be rendered in any modern web browser. Panels at the top switch between various summary figures (e.g. pulse amplitude as function of time), individual event data and if present simulated and cosmic-ray related data. This example shows the electric field and a fraction of the channel waveforms for illustration. The EventBrowser allows for zooming in and out of very plot individually and figures in different tabs are connected for event selection. Figure and caption from [12].

7. Standard processing modules

NuRadioReco comes with a large number of processing modules that cover typical steps of the detector simulation, data processing and low-level event reconstruction of radio detectors. A full simulation and reconstruction cycle for cosmic rays is presented in Fig. 2 which is also available as example in the NuRadioReco repository. This example reads in a CoREAS simulation, performs a full detector simulation to obtain the raw voltage traces, adds noise to the signal, simulates the trigger, and performs a full reconstruction to recover the electric-field radio pulse.

8. New reconstruction algorithms for the electric field

We developed a new *forward folding* technique to recover the incident electric field from the voltage measurement of multiple antennas. The electric field $\mathscr{E}^{\phi,\theta}$ relates to the voltage output $\mathscr{V}_i$ of antenna i in Fourier space as

$$\begin{pmatrix} \mathscr{V}_1(f) \\ \mathscr{V}_2(f) \\ \cdots \\ \mathscr{V}_n(f) \end{pmatrix} = \begin{pmatrix} \mathscr{H}_1^\theta(f) & \mathscr{H}_1^\phi(f) \\ \mathscr{H}_2^\theta(f) & \mathscr{H}_2^\phi(f) \\ & \cdots \\ \mathscr{H}_n^\theta(f) & \mathscr{H}_n^\phi(f) \end{pmatrix} \begin{pmatrix} \mathscr{E}^\theta(f) \\ \mathscr{E}^\phi(f) \end{pmatrix} , \tag{8.1}$$

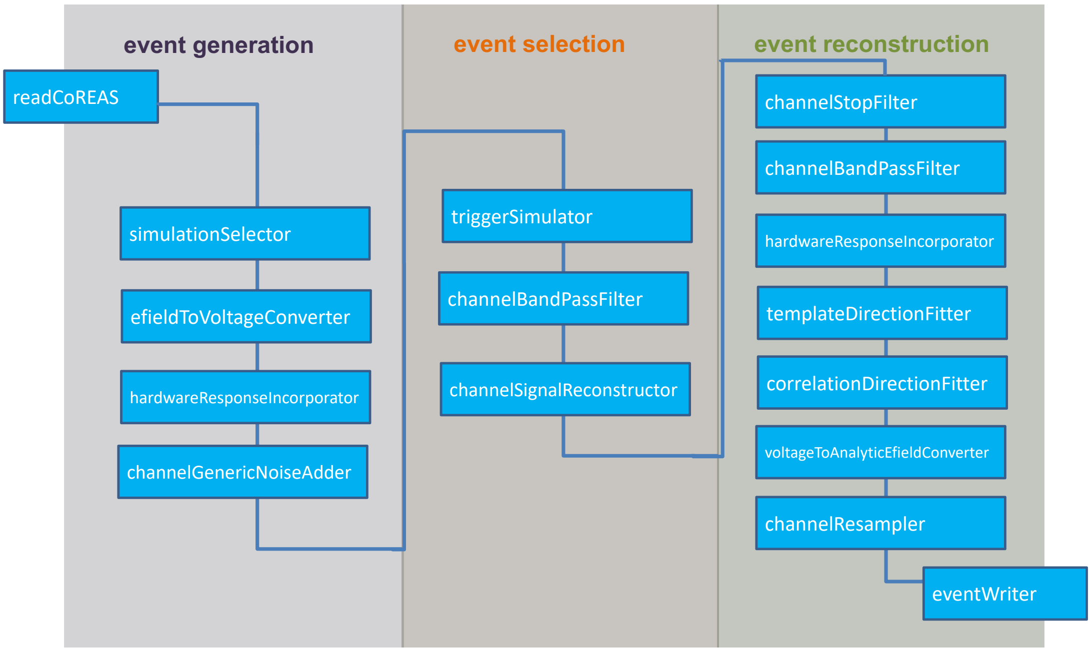

Figure 2: Schematic overview for a full reconstruction cycle for an event.

where $\mathscr{H}_i^{\theta,\phi}$ represents the response of antenna i to the ϕ and θ polarization of the electric field $\mathscr{E}^{\theta,\phi}$ from the direction (φ_0, ϑ_0).

The new technique solves the two main problems of the current standard way of recovering the electric field by unfolding the antenna response, i.e., solving the system of equations of Eq. (8.1) for $\mathscr{E}$: First, for low signal-to-noise ratios (SNRs) the noise gets amplified leading to an overestimation of the electric field. Second, even at high SNR but if only certain frequencies have a low SNR or if one polarization component is not measured directly, the electric field is overestimated as well, which is a common problem in the measurement of horizontal air showers with only horizontally polarized antennas [13].

The *forward folding* technique solves these problems. The cosmic-ray radio pulse can be described well with an analytic function of only four free parameters. This analytic model is (forward) folded with the antenna response (Eq. 8.1) to obtain a prediction of the voltage output of the antennas. These voltage traces are compared with the measurement and the optimal parameters describing the electric field are determined via a χ^2 minimization. This leads to an accurate reconstruction of the electric field (as shown in Fig. 3) and to a substantial improvement in the reconstructed polarization and frequency spectrum (see [12] for details).

This technique is then the basis for the reconstruction of the cosmic-ray energy and polarization [14, 15].

9. Conclusions

NuRadioReco is a modern, modular, Python-based framework for the detector simulation and reconstruction of radio detectors for cosmic rays and neutrinos. It ships with a large number of processing modules that cover all standard processing steps. A new technique to recover the electric

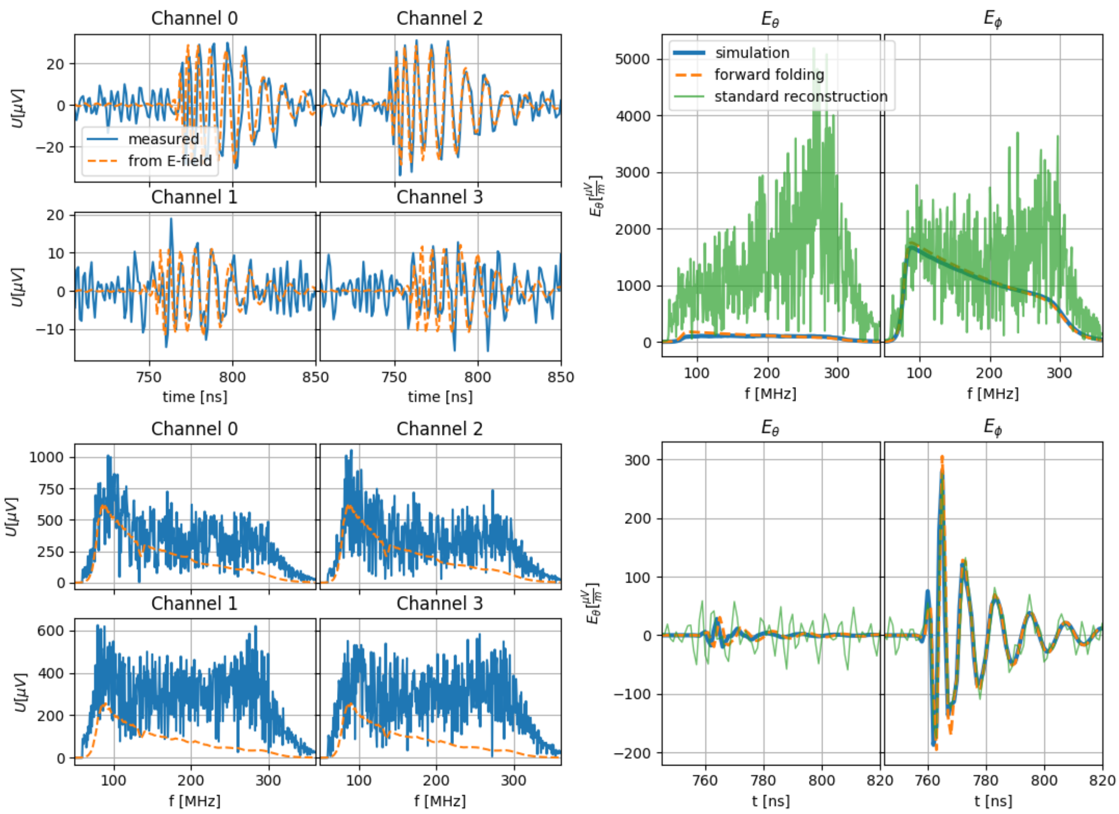

Figure 3: Example of the electric-field reconstruction using the standard and the forward folding technique. Left panels: Voltage traces of four spatially displaces antennas. Shown are both the time- (top) and frequency domain (bottom). The solid blue curve represents the measured voltages whereas the dashed orange curve shows the analytic solution of the forward folding technique. Channels 0,2 and 1,3 are parallel, the measured signal only differs in noise contribution. Upper right panels: Reconstructed amplitude spectrum using the forward folding (dashed orange) and standard (solid green) technique in comparison with the simulated truth (solid blue). Lower right panels: Reconstructed electric field trace using both techniques in comparison to the simulated true values (same colors as above). Figure and caption from [12].

field was presented which leads to an improved reconstruction of the polarization and frequency spectrum, and is the foundation of high-level analyses [14, 15]. NuRadioReco is available open source on github.

10. Acknowledgements

We acknowledge funding from the German research foundation (DFG) under grants GL 914/1-1 and NE 2031/2-1, and the U.S. National Science Foundation-Physics Division (grant NSF-1607719).

References

[1] S. Argirò et al., *The offline software framework of the Pierre Auger Observatory*, *Nuclear Instruments and Methods in Physics Research A* **580** (2007) 1485.

[2] P. Abreu et al., *Advanced functionality for radio analysis in the Offline software framework of the Pierre Auger Observatory*, *Nuclear Instruments and Methods in Physics Research A* **635** (2011) 92.

[3] P. Schellart et al., *Detecting cosmic rays with the LOFAR radio telescope*, *Astronomy & Astrophysics* **560** (2013) A98.

[4] H.-P. Bretz et al., *A development environment for visual physics analysis*, *Journal of Instrumentation* **7** (2012) T08005.

[5] S. Buitink et al., *A large light-mass component of cosmic rays at* $10^{17} - 10^{17.5}$ *electronvolts from radio observations*, *Nature* **531** (2016) 70âĂŞ73.

[6] A. Aab et al., *Energy estimation of cosmic rays with the Engineering Radio Array of the Pierre Auger Observatory*, *Physical Review D* **93** (2016) 122005.

[7] A. Aab et al., *Measurement of the Radiation Energy in the Radio Signal of Extensive Air Showers as a Universal Estimator of Cosmic-Ray Energy*, *Physical Review Letters* **116** (2016) 241101.

[8] S. W. Barwick et al., *Radio detection of air showers with the ARIANNA experiment on the Ross Ice Shelf*, *Astroparticle Physics* **90** (2017) 50.

[9] C. Glaser, M. Erdmann, J. R. Hörandel, T. Huege and J. Schulz, *Simulation of radiation energy release in air showers*, *Journal of Cosmology and Astroparticle Physics* **9** (2016) 024.

[10] A. Nelles et al., *The radio emission pattern of air showers as measured with LOFAR – a tool for the reconstruction of the energy and the shower maximum*, *Journal of Cosmology and Astroparticle Physics* **5** (2015) 018.

[11] NuRadioReco, https://github.com/nu-radio/NuRadioReco.

[12] C. Glaser, A. Nelles, I. Plaisier, C. Welling, S. W. Barwick, D. Garcia-Fernandez et al., *NuRadioReco: A reconstruction framework for radio neutrino detectors*, *The European Physical Journal C* **79** (2019) [1903.07023].

[13] A. Aab et al., *Observation of inclined EeV air showers with the radio detector of the Pierre Auger Observatory*, *Journal of Cosmology and Astroparticle Physics* **2018** (2018) 026.

[14] C. Welling, C. Glaser and A. Nelles, *Reconstructing the cosmic-ray energy from the radio signal measured in one single station*, *submitted to JCAP* (2019) [1905.11185].

[15] A. Nelles for the ARIANNA collaboration, *Cosmic-ray detection with and novel reconstruction algorithms for the arianna experiment*, *PoS(ICRC2019)366 (this proceedings)* (2019) .