A Simulator to Generate VLBI Baseband Data in Matlab

J. Gruber, J. Böhm, A. Nothnagel, M. Schartner

Abstract VLBI radio telescopes represent a sophisticated interconnection of electronic devices. The final product of each observation by a radio telescope is the baseband data. In this presentation, we show a software tool as part of the Vienna VLBI and Satellite Software (VieVS) which generates simulated VLBI baseband data. This simulator is written in Matlab and contains a telescope model which is based on the parametrization of real antenna key characteristics. Moreover, source characteristics such as the signal structure from satellites as well as spatial velocities of sources and receivers are taken into account in the model. Advanced Digital Signal Processing (DSP) algorithms of Matlab are applied to model the antenna system. We will present the various model components of the simulator and discuss potential applications of such a tool. Finally, we show the first results of simulation experiments with the processing in a real VLBI processing chain.

Keywords Baseband data simulation · Correlation

Jakob Gruber, Johannes Böhm, Matthias Schartner
Technische Universität Wien, Department of Geodesy and Geoinformation, Gußhausstraße 27-29/E120, AT-1040 Vienna, Austria

Axel Nothnagel
Universität Bonn, Institut für Geodäsie und Geoinformation, Nußallee 17, DE-53115 Bonn, Germany

(Correspondence: jakob.franz.gruber@geo.tuwien.ac.at)

1 Introduction and motivation

In Very Long Baseline Interferometry (VLBI), baseband data are referred to as the filtered, down-converted, sampled, and quantized electric field strength measurements at each telescope. They represent the final output of each station and act as input data for the correlator to initialize raw data processing. The digital baseband data file contains the time-tagged stream of samples recorded at each station. They are cross-correlated to determine the fundamental VLBI observable. In this work, we present a software tool written in Matlab, which generates such data streams based on the simulation of the VLBI observation process.

The simulator contains a telescope model which is based on the parametrization of real antenna key characteristics and a source model that allows the generation of a specific signal structure. The model parameters are described in Chapter 3 and the systematic, theoretical analysis of the methods are presented in Chapter 2. Baseband data are usually stored in a specific format with a few commonly used format descriptions available. In Chapter 4 the implementation of such a format converter (formatter) is shown.

Existing baseband data simulators have already been successfully used for various studies. The program called 'enoise' (Nishioka, 2015) developed by the VLBI group of Academia Sinica Institute of Astronomy and Astrophysics (ASIAA) has used their simulator to test the zoom band mode of the Distributed FX (DiFX) correlator (Deller et al., 2007). The simulator 'Datasim' (Meyer-Zhao, 2018), which is part of the current DiFX software package, was used for the simulation of higher sampling rates for the Atacama Large Millimeter/submillimeter Array

(ALMA). Both simulators use functions implemented in DiFX and apply them in a reverse engineering way. In contrast, the presented baseband data simulator is developed from scratch and can serve as an independent data source to test correlation algorithms and new observing modes.

The software design of this simulator is intended to be flexible in terms of the source and the antenna model. With this, various types of input signals can be modeled and simulations to artificial satellites with specific signal structures can be carried out. In general, due to the flexible model design and the toolboxes offered by the built-in advanced signal processing routines of Matlab, it is possible to be variable with respect to the simulation of new VLBI observation scenarios (e.g., signal structures, new observing modes). Using the formatter to link the simulated baseband data sets to established correlators, it is possible to pipe simulated data through the real VLBI processing chain. This enables proofs of concepts and evaluating the technical feasibility of simulated observation scenarios.

The development of algorithms for a baseband data simulator implies the inverse mathematical formulation as they are required for certain operations within the correlation process. That applies in particular for the alignment of the telescope data streams due to the difference in signal arrival times and the fringe-rotation due to the Doppler shift. This kind of knowledge gain helps our group to get a better understanding of the operations within the correlator and might be helpful for a more significant interpretation of the correlator feedback. Debugging the observation process to infer a certain antenna behavior from the correlator response might also be another potential field of study.

In this paper, we focus on the description of the computational concept. It provides a general overview of the realization of an independent and flexible baseband data simulator written in Matlab. In analogy to the first fringe detection, the Chapter 5 'First Light' describes the fringe detection of a first basic observation simulation with the presented baseband data simulator.

2 Methodology and realization with Matlab

The baseband data simulator is implemented with Matlab and makes use of the built-in digital signal processing toolbox. The digital signal creation for any sky-frequency is carried out at baseband. The down-conversion from sky-frequency to baseband is not simulated because two times the sky-frequency would be required for the sampling rate. This means at least a sampling rate of 16 GHz would be needed for a signal composed of frequencies in the low X-band at around 8 GHz. A signal simulation of a rather short simulation time of one second would already require 16×10^9 samples, which would exceed the internal memory capabilities of currently used computers. Due to the linearity of the down-conversion it is also possible to carry out any kind of signal manipulation at baseband without degrading the simulation process. To apply and simulate signal manipulation arising from any kind of electronic device within the real antenna signal chain, so-called transfer functions are used. A transfer function is a mathematical representation of the signal manipulation in the frequency domain and describes the relation between the output from the device for possible inputs. The main computational concept of the presented simulator is contained in the formulation of a transfer functions for the antenna system with its individual electronic devices. The transfer functions are applied on the input signal to account for any signal delay, amplitude modification, and phase distortion. The various simulation and modeling possibilities which are currently implemented are described in Chapter 3.

Currently, the Matlab function *randn* is used to generate normally distributed random numbers which are used to simulate source noise and receiver noise. A transfer function with a linear phase term is used to delay the source signals by the difference of signal arrival times between two telescopes. The Doppler shift is modulated in the time domain using the Hilbert transform. The a priori values for the difference in signal arrival times and the telescope velocities are calculated with the Vienna VLBI and Satellite Software (VieVS) (Böhm et al., 2018). The amplitude scale for source and receiver noise is defined by the fundamental relation of signal power with the variance of Gaussian noise. The built-in Matlab Fast Fourier Transformation *fft* is used to switch between the time and the frequency domain. All operations in the simulator are performed using double precision. In the last step the signal with 64-bit amplitude precision is quantized by the number of bits defined in the model parametrization (see Chapter 3).

3 Model parameters

A variety of model parameters is used to characterize the VLBI observation process within this baseband data simulator. There are variables to parametrize the source and the antenna model. Further specifications with respect to other noise components, the observing mode, and the observing geometry can also be made. The source model can be specified by the signal type of the source, either Gaussian noise or signals with a certain structure (e.g. Differential One-way Ranging tones). The normalized amplitude of the selected signal is then scaled proportionally by the specified received signal strength. It is characterized by flux density or antenna temperature values. Besides the source model, the simulator also consists of an antenna model to account for signal filtering and manipulation. Sensitivity parameters in terms of the noise level within the receiver can also be simulated choosing real antenna key parameters like the System Equivalent Flux Density (SEFD) or system temperature (Tsys). A passband filter can be designed with a built-in Matlab toolbox to account for the amplitude and phase response of the total antenna system and for the individual electronic components. Phase distortion can be modeled by specifying a variance for phase noise. Other noise components which affect the antenna system temperature can be modeled by setting individual noise temperatures for the cosmic microwave background, ionosphere and radio-frequency interference (RFI). To define the observing mode, the sampling frequency or observed bandwidth, the number of bits for the digital signal quantization and the number of channels with respect to the sky frequency can be selected. The observation duration can also be defined for each station independently. To consider the observing geometry, a delay accounting for the difference in signal arrival times can be simulated and the Doppler shift due to the relative velocity between the source and antenna platforms can be modeled. Finally, the date of observation needs to be specified to link the observation to a point in time. A more comprehensive list of the model parameters within the presented baseband data simulator can be found in Table 1.

Table 1: Parameter names and their parametrization to configure the presented baseband data simulator

Name	Parameterization
Signal type	Gaussian noise
	User defined signal structure
Received signal strength	Flux density (Jy)
	Antenna temperature (K)
Sensitivity	SEFD (Jy)
	Tsys + effective telescope area (K+m^2)
Sky frequency	(Hz)
Passband filter design*	Ideal
	User defined
Phase calibration signal	Tone frequency spacing (Hz)
Phase distortion	Variance (rad)
System delay	(s)
Polarization	Circular, linear**
Further noise components	(K)
Observation duration	(s)
Sampling	Sampling frequency (Hz)
	Bandwidth (Hz)
Number of bits	(Integer value)
Number of channels	(Integer value)
Group delay	(s)
Velocity station 1	(s/m)
Velocity station 2	(s/m)
Velocity source	(s/m)
Date	YYYY/MM/DDDD/hh/mm/ss
Source name	String
Station name 1	2-letter ASCII code
Station name 2	2-letter ASCII code

* filter for each electronic device
** future implementation

4 Storage format and simulation pipeline

In a final step, the filtered, down-converted, sampled and the quantized data stream is converted to a data format which can be used for further processing at the correlators. Such a format converter is referred to as formatter and represents one of the key components for the presented simulator. It is capable to provide the simulation output in a convenient format which can be used for correlation in the same way as for real baseband data. The same VLBI processing chain consisting of correlation, fringe-fitting, post-correlation processing, and analysis can be fed with either real or simulated baseband data, which allows for profound comparisons and enables various correlation studies. Such a formatter is implemented in Matlab as part of the presented baseband data simulator. The used format is the VLBI Data Interchange Format (VDIF), which can be read by commonly used correlators such as the Distributed FX (DiFX) correlator. The basic VDIF structure consists of data frames, each containing a short self-identifying data frame header, followed by a data array (containing the actual samples). The Matlab formatter is capable to play back the VDIF format and also supports multi-channel recording and recording of several bits per sample. Once the simulated baseband data is stored in the VDIF format, it can be used for further processing. In this work we use DiFX and the Haystack Observatory Postprocessing System (HOPS) to correlate and carry out a delay estimation of the simulated baseband data sets. Besides the VDIF database, also a simple but sufficient VLBI Experiment (VEX) file is generated automatically. It matches the simulated observation parametrization and can be easily used for the DiFX correlation and further processing. Table 2 shows the software packages and databases used in the realization of our simulation pipeline.

Table 2: Software packages and databases involved in the simulation pipeline.

Model parametrization	ASCII file
Baseband data simulator	Matlab program
Baseband data sets	VDIF format
Correlation	DiFX correlator
Fringe-fitting	fourfit

5 First Light

The presented baseband data simulator is the first software solution written in a higher level programming language which is independent of computation algorithms of existing correlators. Figure 1 shows the result of a simple observation simulation and serves as a test object for a basic evaluation of the algorithms within the simulator.

The simulation is based on a scenario with a radio source radiating Gaussian noise observed by two telescopes with realistic SEFD values of 3000 Jy at both telescopes. The telescopes are placed at the same location to isolate the results from Earth rotation effects and large baselines delays. This concept is referred to as zero-baseline observation. Results from non-zero-baseline observations are not described within this work and will be subject to other studies. The rather short observation duration for the presented test simulation is set to two seconds to work around memory issues. One bit is used for quantization and a sampling rate of 32 MHz is applied. The sky frequency is set to 8212.99 MHz. In order to compensate for the short observation time but to secure a fringe-detection the source flux is increased up to 150 Jy. The fourfit SNR estimation after the correlation with DiFX yields a value of 201 while the apriori SNR calculation using the model parameters yields a value of 221. Taking into account the sophisticated SNR calculation with an uncertain determination of the efficiency factors, the values are in a good agreement, which indicates a proper amplitude scaling for the simulated receiver and source noise. A delay of -123 ns is applied for Stat1 to evaluate the implemented delay application method. The estimated delay yields a value of -121 ns. This comparison indicates a proper application of the linear phase term in the simulation process. The difference of 2 ns might be due to a different fringe-reference time which shifts the delay to another point in time, whereas the delay rate absorbs the remaining difference. An ideal passband filter design is applied for this simulation to model the antenna frequency response. The amplitude and phase spectrum of the correlation function also shows a flat behavior which indicates that all applied algorithms do not manipulate the resulting amplitude and phase spectrum. The drop in amplitude for the last sample might be due to the sampling of the frequency spectrum and cannot be seen for finer spectral resolution within the correlation process.

The comparison between expected and real processed values shows a good agreement in general. The remaining discrepancies can arise from various error sources, which might not specifically be dedicated to the simulator algorithms. The used concept of very short observation time and the zero-baseline observation might have some impact and might cause an unusual behavior within the correlation and fringe-fitting process. With the increased source flux value to compensate for the short observation time, the ratio of the source noise amplitude and receiver noise amplitude gets larger than usual. Some further investigations are required to verify the application of short observation intervals.

6 Conclusions and outlook

The presented basic evaluation of amplitude scaling, delay application, and passband filtering serves as the first proof of concept for the baseband data simulator written in Matlab. It shows that it is possible to realize such an implementation with a higher level program-

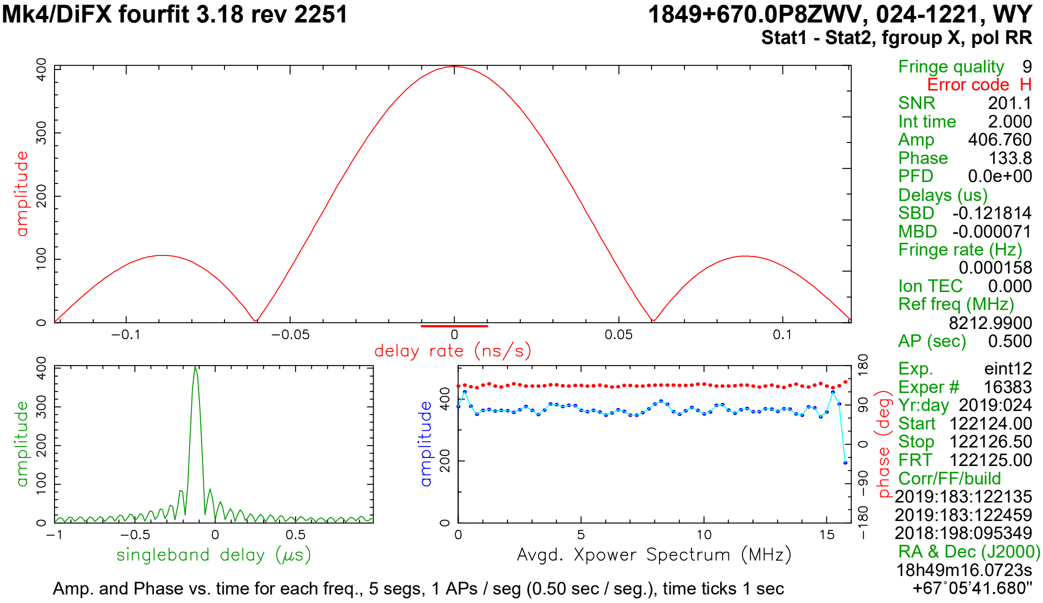

Fig. 1: Fourfit output of simulated baseband data after correlation with the DiFX software. The plot shows the correlation function in the time domain with a distinct peak in green color, the delay rate function in red and a flat amplitude and phase spectrum in blue and red. The simulation setup is configured as follows: two seconds observation time, one-bit quantization, 32 MHz sampling rate, 150 Jy source flux, 3000 Jy SEFD for both telescopes, ideal bandpass filtering, 123 ns delay application.

ming language from scratch without dependencies on existing correlators. The built-in routines *randn*, *fft* and the filter design toolbox prove to be a handy foundation for such a digital signal processing software realization. With VLBI baseband data being characterized by very high data rates, memory issues for any kind of software realization are unavoidable and not related to Matlab. It is a problem for other (proprietary) programming languages as well. Of course, with respect to the run time of the program, other languages will have advantages in contrast to higher level programming languages. However, some effort has been made to apply parallel computing strategies to keep the simulation run time short.

We see a lot of potential application for the simulation of new observing modes and in particular for satellite observations. The simulation pipeline can be used to test correlators with special observation setup and source characteristics. The evaluation of correlation results with short duration and large source amplitude in comparison to commonly used longer observation time seems to be an important field of study for efficient and memory inexpensive simulations. There is still the possibility to use the supercomputing infrastructure at TU Wien, but it would be much more efficient to obtain reliable correlation results with short observation intervals as well.

Future work regarding software development will address the implementation of non-zero-baseline observations and the possibility to simulate linear polarized observations. This could be of great interest in testing the multi-band delay estimation for VGOS observations.

References

Böhm J, Böhm S, Boisits J, et al. (2018) Vienna VLBI and Satellite Software (VieVS) for Geodesy and Astrometry. *PASP*, 130, 044503, doi:10.1088/1538-3873/aaa22b.

Meyer-Zhao Z (2018) Datasim – A VLBI baseband data simulator. *DiFX User and Developer Workshop 2018, Wettzell*

Nishioka H (2015) DiFX activities at ASIAA. *DiFX User and Developer Workshop 2015, Hobart*

Deller A T, Tingay S J, Bailes M, et al. (2007) DiFX: A Software Correlator for Very Long Baseline Interferometry Using Multiprocessor Computing Environments. *PASP*, 119, 318–336, doi:10.1086/513572