



*IMT
EPFL
ASTRON
MPIA Heidelberg
Observatoire de Genève
University of Leiden/NOVA*

Astrometric Survey for Extra-Solar Planets with PRIMA Simulator for testing PRIMA DRS PRIMA Astrometric Observations Requirements and Design

Doc. No. UL-TRE-AOS-15752-0020
Issue 0.3
Date 24 sep, 2004

Prepared . . . Jeroen A. de Jong . . . September 24, 2004 . . .
Name Date Signature

Approved . . . N.N.
Name Date Signature

Released . . . N.N.
Name Date Signature

Change Record

Issue	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
0.1	27-Aug-2004	all	created

Contents

1 OVERVIEW	1
1.1 Scope of the Document	1
1.2 Acronyms	1
2 INTRODUCTION	3
3 REQUIREMENTS	3
3.1 Scientific	3
3.2 Technical	4
3.3 Standard	4
3.4 Interfaces	4
3.5 Calibration	5
4 DESIGN	5
4.1 Configuration files	5
4.1.1 Catalog	6
4.1.2 Schedule	7
4.1.3 Detector	7
4.1.4 Perturbation	7
4.1.5 Flags	8
5 Position and OPD computation	8
6 Application of perturbations	9
7 Detector	9
8 Output files	9

List of Figures**List of Tables**

1 OVERVIEW

1.1 Scope of the Document

- [1] W. Bogo, A. Busso, S. Mottini, *et al.*, “PRIMA FSU design and analysis report,” Tech. Rep. VLT-TRE-ALS-15740-0004, Alenia Spazio, 2003.

1.2 Acronyms

DRL	Data Reduction Library
DRS	Data Reduction Software
OPS	Observation Preparation Software
DTD	Document Type Description
DOPD	Differential Optical Path Delay
OPD	Optical Path Delay
XML	Extensible Markup Language

2 INTRODUCTION

This document describes the requirements and design for the PRIMA simulator. This simulator will provide OBs with simulated data based on an elaborate configuration. The base code of the simulator will produce FSU data from an ideal interferometer without atmosphere. All other effects are considered perturbations which are computed by external modules for which a comprehensive API will be provided.

3 REQUIREMENTS

This section lists the categories of requirements. For each category a basic list of fundamental, and verifiable requirements are listed. If possible each requirements has an minimum requirements listed, and a goal. The requirements are structured as follows:

1. Scientific (Section 3.1)
2. Technical (Section 3.2)
3. Standards (Section 3.3)
4. Interfaces (Section 3.4)
5. Calibration (Section 3.5)

3.1 Scientific

1. The simulator computes the predicted optical pathlength difference between the science and reference source given a number of potential observing conditions.
2. The simulator takes into account errors as a result of:
 - (a) astrophysical errors (e.g. astrometric jitter of reference star)
 - (b) atmospheric errors (e.g. turbulence, refraction, dispersion)
 - (c) VLTI infrastructure (e.g. delay line tunnel weather conditions, non-common path errors)
 - (d) PRIMA instrumental errors (e.g. fringe sensing noise and errors)
3. The simulator calls the modules which introduce perturbations in sequential order. This adds dependencies (synergy) between the modules that could not be obtained from running the isolated modules independently on their own.
4. The simulator should produce a FITS file consistent with IAU standards, and with ESO standards. The FITS file format should be identical to the one adopted by ESO for PRIMA. If IAU and ESO standard contradict, preference should be given to the IAU standard.
5. A typical simulation has the following characteristics:
 - (a) Two sources observed with two ATs;
 - (b) Observed on the two longest baselines (orthogonal): B5-J6: 201.916 m and D0-J2:100.242 m;

- (c) Large magnitude range up to $K=16$;
- (d) Sources may or may not be resolved, depending on their distance.
- (e) The spectral types for both sources are not identical, and both are in range of A to M-type;
- (f) Both sources have a proper motion and parallax;
- (g) The exposure has a duration of ≈ 30 minutes, continuously. Changes of the projected baseline, and elevation should be taken into account;
- (h) The K-band needs to be resolved in at least 10 frequency bins;
- (i) At least two atmospheric conditions should be included in the simulator (e.g. typical winter, typical humid summer);

3.2 Technical

1. The simulator needs proper APIs to communicate with modules written in different programming languages. For some programming languages this may go through an ASCII file interface.
2. The simulator should be designed such that it can become significantly more complex over the years.
3. Considerable effort should be taken to minimize the amount of computing time. Therefore, it is very important to use appropriate approximations.
4. The accuracy to which errors should be specified should be better than 1 nm OPD.
5. Visualization tools are not integrated in the simulator. They are external tools.

3.3 Standard

1. No formal software standards are imposed on the simulator, except that the FITS output file should be IAU and ESO standard, and that if available ESO keywords should be used.
2. The software coding language should have the following features:
 - (a) Allow evolution of the software towards ESO standards
 - (b) Allow use of modules by the OPS and DRS
 - (c) Allow to evolve to a system that is WWW-accessible

3.4 Interfaces

1. The input configuration files will require an hierarchical and extensible format considering the complexity of the configuration and the need to evolve this easily. XML should be considered for this format, since it is a widely accepted standard for which many parsers and tools are available.
2. Modifying the input files will initially go through an standard editor (e.g. vi), but the goal is to do this with a GUI (to allow consistency checking).
3. The configuration files are structured using ESO keywords (when available).

3.5 Calibration

A list of calibrators is needed to allow the consortium to use the first data to train the DRS. Observations of these calibrators will also be simulated to test the trend analysis.

The following list of 13 possible calibrator stars is taken from the high precision subset of the *Sixth Catalog of Orbits of Visual Binary Stars* by Hartkopf & Mason (2004) recommended for calibration purposes. Only systems with separations between $2''$ and $10''$, and with declinations between -70° and $+20^\circ$ were selected. The format of the table is the same as in the original catalog.

RA,Dec (2000)	WDS	Name	ADS..	HD...	HIP...	V1...*	V2...*	...P.....	...a.....	...i....	Omega...*	...T.....	...e... omega...	Referenc	G	N	P	E	
013947.24-561147.2	01398-5612	DUN 5AB		10361J	7751	5.82	5.86	483.66	7.81689	142.824	13.116*	1813.494	0.5344	18.374	vAb1957	5		P	E
020202.80+024549.5	02020+0246	STF 202AB	1615	12446	9487	3.82	4.92	933.05	4.0	120.9	23.3	2098.643	0.696	225.4	Sca1983f	4	N	P	E
033647.31+003517.4	03368+0035	STF 422	2644	22468	16846	5.82	8.69	2101.	8.023	32.07	92.35	1900.0	0.18	151.97	Hop1964b	5	N	P	E
042316.83+112240.8	04233+1123	STF 535	3174	27762	20472	6.64	8.01	1128.00	2.040	145.5	114.0	2046.00	0.495	243.6	Hrt2000c	5	N	P	E
050029.67+050556.6	05005+0506	STT 93	3596	32022	23277	8.02	8.68	1467.52	2.530	99.7	241.3	1914.22	0.649	249.3	Sey1999a	5		P	E
081212.71+173853.3	08122+1739	STF1196AB-C	6650	68257	40167	5.05	6.20	1115.	7.70	146.	74.2	1970.	0.24	345.5	Hei1996b	4	N	P	E
084646.65+062508.1	08468+0625	STF1273AB-C	6993	74874	43109	3.38	6.48	990.	4.66	39.	49.3	* 1920.	0.30	200.	Hei1996b	4	N	P	E
121558.52+053825.2	12160+0538	STF1621	8486		59816	9.45	9.96	1100.	3.5	26.	2.	1973.67	0.89	275.	Sod1999	5	N	P	E
123033.75+094257.2	12306+0943	STF1647	8575	108875	61035	7.46	7.72	4273.	4.358	67.0	221.5	1864.3	0.6900	341.8	Hop1970	5	N	P	E
135458.30-080331.6	13550-0804	STF1788AB	9053	121325	67953	6.19	6.97	2613.	4.998	50.9	108.05	1823.8	0.35	293.5	Hop1970	5	N	P	E
161556.96+072129.6	16160+0721	STF2026	9982	146413	79702	8.70	9.08	473.5	2.457	133.1	12.3	1906.9	0.82	192.5	Hei1963a	4		P	E
180527.21+023008.8	18055+0230	STF2272AB	11046	165341	88601	4.03	5.87	88.38	4.554	121.16	302.12	1895.94	0.4992	14.0	Pbx2000b	1	N	P	E
204613.25+155425.8	20462+1554	STF2725	14270	197913	102490	7.09	7.90	2851.	7.364	66.0	20.9	1658.	0.18	276.3	Hop1973b	5		P	E

4 DESIGN

The simulator consists basically of modules for the following tasks:

1. Configuration file reader and setup.
2. Computation of target positions and ideal OPDs between the target and reference star beams.
3. Applying various time and wavelength dependend perturbations on the OPDs. These perturbations can be due to:
 - Astronomical errors
 - Atmospheric errors
 - VLTI errors
 - PRIMA errors
4. Computing the FSU detector output.
5. Writing the files of each OB in the PRIMA FITS format with the computed detector output. These files must mimick a true set of raw PRIMA data, including other data sources besides the FSUs.
6. Writing files with processed data which only contain the differential OPDs.

Module 4 is optional and can be skipped when only reduction steps after the detector data reduction have to be tested. In that case module 6 will be used to write the output data.

4.1 Configuration files

The configuration file should describe everything for constructing realistic OBs, which means that they should allow for:

- Specifying several targets and calibrators
- Providing observation schedules
- Providing all physical information about the targets
- Specifying the telescope/baseline configuration
- Specifying which perturbations should be computed and providing the corresponding modules with the necessary parameters.

The configuration file will have a hierarchical structure, which makes a simple keyword value approach impractical. A very commonly used format for such configuration files is XML. In this format context dependent tags are used which follow the same format as HTML tags. An optional DTD describes which tags are allowed in each context and what their attributes are. The main advantage of XML is that there many parsers available for all major programming languages. This saves time for inventing our own format and writing a parser for it.

The general syntax of XML is as follows:

```
<tag1 attr1="a" attr2="b">
  GENERIC CONTENT
  <child_tag1 attr2="1" attr3="2"/>
  <child_tag2 attr4="NONE">
    GENERIC CONTENT
  </child_tag2>
</tag2>
```

So, opening tags are always denoted with `<tag>` and closing tags always with `</tag>`, where tag can be replaced by any descriptive text, not containing spaces. Inside the tags space delimited attributes may be added. In the following sections the tags are given with their parent tags within brackets. The possible attributes are given in *italic* in the descriptions. Generic content is denoted with a bold "CONTENT".

In the following sections a preliminary version of the configuration file is described. The sections can be split among different files which may combined on the command line of the simulator. This allows for e.g. a single catalog with different schedules.

4.1.1 Catalog

catalog Describes the catalog of targets.

name Name of the catalog.

description(catalog) Describes how the catalog is formatted. This is used to interpret the further contents between the `<catalog></catalog>` tags.

delimiter Delimiter between the columns. If omitted the widths of the columns must be given.

column(description) Describes one column in the catalog.

name Name of the column

type Type of data (STRING,FLOAT,INTEGER)

width Width of the column (optional)

CONTENTS: The ASCII catalog data.

4.1.2 Schedule

schedule Describes an observing schedule.

name Name of the schedule.

OB(schedule) Describes an observation block. This block consists of several exposures.

exposure(OB) Describes a single exposure.

target Target name

start Start time (YYYY-MM-DD HH:MM:SS.S)

ndit Number of integrations

dit Detector integration time

setup Identifier of an instrument setup.

setup(schedule) Optional specific keywords for an instrument setup.

iden Identifier of the setup. Must correspond with the identifiers in the setup child tags of the exposure tags. **CONTENTS:** List of ESO hierarchical keywords. Each line has a **key = value** format.

4.1.3 Detector

detector Describes the detectors. This information is required for constructing the simulated image data.

nx Number of pixels in X direction

ny Number of pixels in Y direction

ron Read-out Noise

pixsens(detector) Individual pixel sensitivities. Contains an array of floating point numbers.

specsens(detector) Global spectral sensitivity. Contains an array of floating point numbers.

wstart Start wavelength

deltaw Wavelength step

nvalues Number of values in array

4.1.4 Perturbation

The perturbation sections of the configuration file describe the interfaces between the core of the simulator and the various external modules which compute these perturbations. The external module will at least get access to all information computed by the simulator so far. Additional parameters can be set in the configuration file as described below.

perturbation Interface to perturbation algorithm

name Name of the perturbation

interface Corresponding interface class

required_info(perturbation) Required information for computing the perturbation. *name* Name

type Data type (string,float,integer,array)

value Value

4.1.5 Flags

This section describes how sophisticated the simulation should be. If this section is empty or missing data for an ideal interferometer without atmosphere will be computed.

flags Describes which computations should be enabled.

perturbation(flags) Enables a specific perturbation
name Name of the perturbation
use Use this perturbation? (yes/no)

detectordata(flags) Enables the computation of detector data.
use Compute detector data? (yes/no). If no, only differential OPDs will be computed.

5 Position and OPD computation

This module starts with setting up the internal data model for the simulator. The data model will store the simulated data in the following variables (besides the fixed configuration parameters):

t An array of integration timestamps. Obtained from the schedule.

\vec{B} Array of coordinates of the used VLT stations (ATs or UTs). Each coordinate is 3 element vector with respect to the VLT reference coordinate.

$\vec{S}(i, t, \lambda)$ Stokes parameters as function of beam index i , time and wavelength.

$\Delta D(j, t, \lambda)$ OPD for reference star ($j = 0$) and target star ($j = 1$)

$\vec{C}(j, t, \lambda)$ Apparent right ascensions, declinations, altitudes and azimuths for both stars

The wavelength should be sampled well enough to compute a realistic fringe pattern for each of the 3 PRIMA wavelength channels.

The ideal simulated data will be computed as follows:

1. for each target the apparent coordinates at the time of observation are computed by applying the following corrections:
 - Parallax
 - Proper motion
 - Orbital elements of binaries if applicable
 - Precession
 - Nutation
 - Aberration due to the observers velocity with respect to the target.
2. the altitude and azimuth on the sky and the geometric OPDs.
3. the ideal differential OPD as measured outside the atmosphere with respect to the first target.

All these values do not depend on wavelength.

6 Application of perturbations

The simulator calls external subroutines or interpolates from tabular information in order to compute a perturbation on the OPDs as function of time and wavelength. For embedding external subroutines written in C, C++, Java or Jython an API will be provided. This API provides:

- access to all information within the simulator, being:
 - all parameters from the configuration file
 - the arrays containing the simulated data (see previous section)
- return stub for perturbations on ΔD , \vec{S} and \vec{C}

The external subroutines compute the ΔD , \vec{S} and \vec{C} values for a given integration timestamp. They may store intermediate data in the API class to improve the computational efficiency.

If the user wishes to compute the perturbations in another language then an ASCII table can be provided, from which the simulator will interpolate to obtain values at the specified timestamp. If the provided time range of the table does not cover the schedule then the simulator will repeat the same values in another time block. This can be useful for e.g. atmospheric simulations to save computation time.

7 Detector

Simulates the interference of the perturbed wavefronts in the beamcombiner and fringe measurement by the FSUs. This module must follow the principles of the FSU design. AD 1 The FSU samples the interferometric fringe at 4 phases separated by $\pi/2$ and at 3 wavelength channels. The simulator should compute the interference pattern from the ΔD and \vec{S} . This requires taking into account polarization since the FSU has a component which introduces a 90° phase shift between the two components of the polarization vector. Later the beam is splitted by polarizing beamsplitter to produce two beams with this 90° phase shift. If the incoming beams have already some degree of polarization this will influence the relative power in the channels on the detector.

8 Output files

Creates for each exposure in the schedule a FITS files for which the format must fully conform to the standards of the real PRIMA observations. This format is currently under construction, so the code of this module should be flexible enough to deal with frequent changes. Also the contents of the data will depend on which input sources should be simulated.

---oOo---