WRS documentation ver. 0.9

28 Apr 2014
Marco Stangalini
INAF/OAR Osservatorio Astronomico di Roma
marco.stangalini@oa-roma.inaf.it

This document is a short guide for the use of WRS (Wavefront reconstruction software) and for its first tests on LBC. WRS is still under development and any bug report or comment will be appreciated.

Dependencies:
WRS is completely written in IDL and needs Astrolib to run.

WRS is currently provided in two forms:
- a standalone version
- a LBCFPIA integrated version

Quick guide

In order to compile the required routines, the command:

@make_WRS

must be given before starting the WRS execution. This must be done before running both standalone and LBCFPIA integrated version. In what follows we assume that all the WRS routines are in the IDL working directory.

After setting up the WRS parameters in the *WRS.par* file, WRS can be launched. Depending on the configuration chosen, many information about the execution are stored in the *WRS_report.dat* file. In addition some information and plot can be also displayed on the screen if needed.
If the keyword SAVE_WRS is enabled in the configuration file, then the selected pupils, the final model obtained through WRS, and all the results are saved in the SAVE_DIR directory.

Standalone version

In the first case WRS can be used manually on a generic fits image that can be directly provided as input of the code itself.
The main function to do this is **STANDALONE_WRS.pro**.

USAGE:

STANDALONE_WRS, filename

In this case no correction is sent to the telescope, and the Zernike coefficients are printed on the screen and in the log file (see following section for more information on the log file).
In case one needs to send the coefficients to the telescope using *SendTCS* it is sufficient to use the command:

STANDALONE_WRS, filename, /SendTCS

in this case, the Zernike coefficients will be saved on a file using the WriteOutFile routine (already used in LBCFPIA) and then sent to the TCS using TCSsend tool.

LBCFPIA integrated version

WRS can be also used integrated into LBCFPIA through the routine **LBCFPIA_WRS.pro**. In this case the previously

used routines CalcAbCoeffs.pro and FindPupils.pro have been substituted by the function CALL_WRS.pro, which now is in charge of finding the pupils in the input image and calling WRS for the analysis of the aberrations.

In this case, LBCFPIA can be run the usual way and the only difference (hidden to the end user) is represented by the method used for the estimation of the aberrations.

However, many parameters already present in LBCFPIA and set up for the old configuration after many tests remain the same. This means that some of them may not be suited for WRS and may become the cause of errors which are difficult to identify. For this reason it is highly recommended to only use LBCFPIA_WRS after a first manual test of WRS through the standalone version. This may help identifying bugs in the WRS core or in the LBCFPIA interface.

Please note that due to the lack of TCS and other subsystems, the LBCFPIA_WRS routine has not been extensively tested and debugged. For this reason some (hopefully small) bug might be present.

The configuration file WRS.par and the WRS usage

In the configuration file WRS.par all the WRS relevant parameters can be set up. Each parameter is described directly in the WRS.par file:

```
;
================================================================================================
========
;WRS PARAMETERS
;The following parameters let the user set up WRS, the wavefront reconstruction code for LBC.

;For information or to report bugs please contact marco.stangalini@inaf.it
;first version: Apr 2014

;
================================================================================================
========


{wrs,
D:        8.40000,              ;Telescope diameter in [m]
EPS:      0.2200,               ;Telescope obscuration [%]
LAMBDA:   0.550000,             ;working wavelength in [mu]
PIXEL:    0.223,                ;pixel scale [arcsec]
NGRID:    1024,                 ;Padding. Do not change unless you know what you are doing
THRESH:   0.4,                  ;Numerical threshold used in the WRS code. Do not change unless you know what you are doing
NZER:     12,                   ;Maximum number of Zernike terms to be estimated. Used only in WRS slow mode. Larger than 12 may coase
                                instabilities.
SMOOTHING: 3,                   ;Smoothing window to reduce noise. 3 is a good option. Do not change unless you know exactly what you are doing.
                                Use with care.
DEFAULT_SIGMA: 3,               ;Starting sigma noise threshold. If there is no pupil above this value, WRS will try to automatically adjust the
                                threshold before giving up.
FAST_MODE: 1,                   ;FAST_MODE=1,0 --> YES, NO. In the fast mode only Zernike up to #6 are estimated. This is very fast (less than 1
s)
DISP: 1,                        ;DISP=1,0 --> Enabled, Disabled. It enables the display of the information about the pupil selected and its relative
                                model
EFOC: -1,                       ;Extra- or intrafocal mode (do not change)
THRESHOLD_ADJ: 0.9,             ;If there is no pupil detected the detection threshold is iteratively lowered. Example: sigma=sigma*threshold_adj
SAVE_DIR: './save/',            ;Saving dir used to save plots and images. Used only if SAVE_WRS is on. The directory chosen must be created
                                before running WRS.
SAVE_WRS: 1,                    ;If eq 1 WRS saves the image and the model of the selected pupil in a sav file. The filename will contain the
                                timestamp as in the log file.
soft_ver: '0.9'                 ;WRS software version
}
```

While most of the parameters are fixed or its meaning is clearly evident, some of them deserves more explanations.
There are two operational modes in WRS: **the fast mode and the slow mode**.
In the fast mode only defocus and astigmatism are estimated. This mode is very fast (less than 2s on a i7 Intel core). The FAST_MODE keyword allows the selection of the required mode depending on the specific needs. The slow mode is generally recommended if one wants to correct the shape of the primary mirror before starting the observations.

The *DEFAULT_SIGMA* parameter defines the detection threshold that must be used in the pupil selection. Only the best pupil in terms of S/N is chosen in the code. If there is no suitable pupil corresponding to the selection criteria, WRS will try to automatically adjust the initial detection threshold until a good pupil is found. This will require less than one second of computation. During the automatic adjustment, the image quality is checked to see if there is a good pupil above the noise level.

If the automatic adjustment fails, WRS will stop complaining about the image quality or the integration time.
For this reason the choice of the *DEFAULT_SIGMA* parameter is not critical. Good values are between 3 and 4. The step using in the automatic adjustment is defined by the parameter *THRESHOLD_ADJ*.

A critical parameter is in turn the padding *NGRID*. This corresponds to the size of the array which is used in the computation, and should be a power of 2. This parameters also dictates the computation time.
If the initial defocus is large, then the computational grid should be large too. One may think that the larger the initial defocus, the better the Zernike estimation. This is true only in an ideal situation when noise in not taken into account. The fact that the S/N ratio is always finite implies that there will be an optimal value of the input defocus which maximizes the accuracy of the estimation.
By default NGRID is set to 1024. This is the minimum value needed to analyze the images acquired with the commonly used initial defocus in the science camera before running the Ob.
A value of 1024 means that the pupils must be entirely contained in at least a FoV of 64x64 pixels, which is usually the case in LBC archive imagery. Some simulation performed to optimize this value have revealed that a good accuracy is also obtained with smaller initial defocus (smaller pupils).

WARNING: it is highly recommended to choose the initial defocus in such a way that the diameter of the resulting pupils is between 30 and about 55 pixels (with NGRID=1024).

It is also recommended to enable the SAVE_WRS mode during tests, in order to have all the information about the WRS easily available for debugging.

NOTE: During the pupil selection process, WRS checks for the presence of overlapping pupils and automatically removes them. When the display of the pupils found in on (DISP parameter in the configuration file) it may happen that some of the selected pupils appear overlapped. This doesn't mean that the pupil selection is not working properly leaving overlapping pupils, but it is only because the boxes used to highlight each pupil are larger that the real case in the plots, depending on the size of the window used and on the screen resolution.

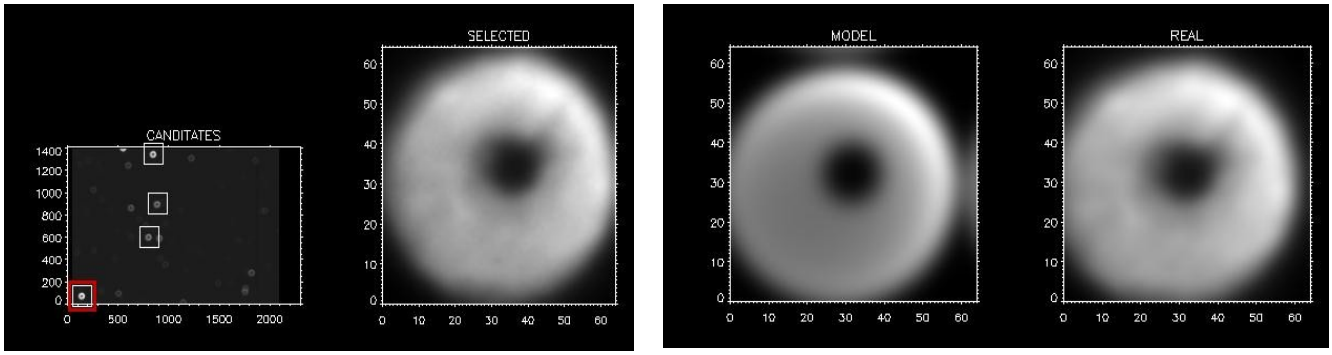

Fig. 1 Example of images saved in case the DISP keyword in the parameter file is enabled.

The log file WRS_report.dat

All the information obtained during the execution of WRS are saved in the WRS_report.dat. In addition, as aforementioned, if the SAVE_WRS switch in the configuration file is enabled, the images of the pupils (Fig. 1) as well as the estimated Zernike and the main arrays are also saved. To help the identification of each .sav file with its corresponding information in the log file, a unique identifier is used in the filename and in the log file itself, indicating each individual WRS correction step. This identifier is the system time.

In the the following we show an example of log file

```
================================================================================================================
================================================================================================================
WAVEFRONT RECONSTRUCTION Tue Apr 29 15:09:42 2014
Parameters are read from ./WRS.par
```

```
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
                    WRS wavefront reconstruction software ver. 0.9
                    Tue Apr 29 15:09:42 2014  WRS called
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
WRS INITIALIZATION...
Rebinning factor:       32
Grid pixel:    0.00696875 arcsec
Grid size:       16.2793 m
CCD pixel:       0.223000 arcsec
FOV field:       14.2720 arcsec
FOV pixels:      64

WRS FAST MODE off -- Display mode on
max number of Zernike to be estimated       12
Searching for pupils...
      4 CANDIDATES FOUND BELOW CCD SATURATION LEVEL
CHECKING AND REMOVING OVERLAPPING PUPILS...
./save/FINDPUP_results_Tue Apr 29 15:09:42 2014.jpg SAVED
BEST SUITED PUPIL @ X,Y       143.925      76.0511
CROPPING IMAGE...
COMPUTING ZERNIKE TERMS...
 Z        1      2      3      4      5      6      7      8      9     10     11     12
Cycle:       1  RMS=       19.8769 percent
 um    0.500  6.164  3.875 20.931  0.361  0.389  0.000  0.000  0.000  0.000  0.000  0.000
Computing the interaction matrix...
Cycle:       2  RMS=       16.5662 percent
 um    0.807  6.178  3.880 20.896  0.333  0.310 -0.047 -0.085 -0.052 -0.007  0.006 -0.010
Cycle:       3  RMS=       14.8332 percent
 um    0.988  6.187  3.882 20.874  0.313  0.258 -0.078 -0.140 -0.087 -0.013  0.009 -0.016
Computing the interaction matrix...
Cycle:       4  RMS=       11.0378 percent
 um    1.474  6.336  3.621 20.866  0.200  0.246 -0.174 -0.234 -0.110  0.054 -0.032 -0.018
Cycle:       5  RMS=       10.0191 percent
 um    1.500  6.414  3.473 20.877  0.176  0.235 -0.231 -0.288 -0.115  0.084 -0.063 -0.019
Computing the interaction matrix...
Cycle:       6  RMS=       8.75524 percent
 um    1.500  6.003  2.914 20.934  0.247  0.245 -0.304 -0.331 -0.042  0.022 -0.137 -0.013
Cycle:       7  RMS=       8.47775 percent
 um    1.500  5.825  2.697 20.968  0.283  0.247 -0.339 -0.351 -0.015 -0.004 -0.174 -0.009
Computing the interaction matrix...
Cycle:       8  RMS=       8.48033 percent
 um    1.500  5.775  2.583 21.012  0.313  0.246 -0.370 -0.364 -0.003 -0.003 -0.205 -0.003
Divergence... Now LM parameter =   1.00000e-06
Cycle:       9  RMS=       8.45892 percent
 um    1.500  5.800  2.646 20.990  0.298  0.246 -0.354 -0.358 -0.010 -0.003 -0.189 -0.006
Fitting done!
./save/WRS_images_Tue Apr 29 15:09:42 2014.jpg SAVED
WRS results saved in: ./save/WRS_results_Tue Apr 29 15:09:42 2014.save
```

Zernike coeffs. estimated [mu]:

Seeing [arcsec], tip, tilt, Defocus, 45 deg ast, 0 deg ast, Y coma, X coma, Y trefoil, X Trefoil, Sph,...

1.50000 5.80042 2.64563 20.9902 0.298200 0.246131

-0.708277 -0.715110 -0.0194249 -0.00635234 -0.378647 -0.0116761

WRS iteration completed onTue Apr 29 15:10:31 2014

==========

List of the main routines and files

@make_WRS
It compiles the necessary routines. It must be executed before running WRS or LBCFPIA_WRS

standalone_WRS.pro
It start the WRS analysis and the pupil selection on a fits image

CALL_WRS.pro
Is the main function invoking WRS
Input: filename of the fits image to be processed
Output: Zernike coeffs

LBC_WRS.pro
Is the core of the WRS analysis. It is called by CALL_WRS and starts the pupil selection and the wavefront reconstruction

WRS_LIB.pro
It contains the main functions needed to estimate the Zernike coeff. It is compiled by @make_WRS.

WRS.par
It is the main configuration file

WRS_report.dat
The log file