Very Large Telescope

Paranal Science Operations

NACO data reduction cookbook

Doc. No. VLT-MAN-ESO-14200-4038

Issue 81.0, Date 25/12/2007

O. Marco, G. Chauvin, N. Ageorges

Prepared . . . .
                                                    Date   Signature

O. Hainaut

Approved . . . .
                                                    Date   Signature

A. Kaufer

Released . . . .
                                                    Date   Signature

This page was intentionally left blank

Change Record

Issue/Rev.	Date	Section/Parag. affected	Reason/Initiation/Documents/Remarks
1.0	1/07/2006	all	Creation by Olivier Marco
2.0	15/08/2006	Sect. 7	Updated by Gael Chauvin
3.0	15/09/2006	all	First offical release by Nancy Ageorges
81.0	25/12/2007	all	Update for P81 release (update about CPL ...) - NAg

This page was intentionally left blank

Contents

1 Introduction

1.1 Purpose

The document is intended for astronomers who want to reduce NACO data. It describes the various data formats delivered by NACO, observational scenarios and reduction procedures. This document concentrates on the methodology rather than individual routines that are available in either IRAF or MIDAS. The document describes the algorithms implemented in the eclipse data reduction package. The ECLIPSE package is an integral part of the NACO pipeline, which produces calibration products, quality control information and reduced data. The pipeline does produce reduced data; however, this is not meant to replace more general reduction packages such as IRAF, MIDAS or IDL. The pipeline does not replace interactive analysis and cannot make educated choices. Thus, the data that the NACO pipeline produces should be considered as a way of quickly assessing the quality of the data (a quick look if you like) or a first pass at the reduction of the data. Throughout this document we will list the eclipse routines that are used to reduce NACO data and we will give a short description of how they can be used. We will also list the shortcomings these routines have, so that users can decide if they need to reduce their data more carefully. For completeness, we have also included a description of the eclipse routines whose primary aim is to provide quality control information. These routines are probably of little interest to astronomers. This document does not describe the NACO instrument, its modes of operations, how to acquire data, the offered templates, or the various issues attached to Phase II Proposal Preparation. The reader is assumed to have read the NACO User's Manual beforehand, and have a basic knowledge of infrared data reduction in imaging and spectroscopy. This document is a living document, and as such follows the evolution of the recipes, the implementation of new algorithms, enhancements, new supported modes, etc. Anything related here has to be understood to be valid at the date of writing.

Since this document has been created, the pipeline has been updated and now works under CPL (=Common Pipeline Language). The information contained in this manual are nevertheless still valid and definitively useful, since the current implementation is based on the same 'scientific data reduction recipes'. Hereafter for details about reduction scripts (recipes) mentioned, we refer you to the pipeline user's manual.

1.2 Reference documents

1. ESO DICB - Data Interface Control Document - GEN-SPE-ESO-00000-0794
2. NACO Calibration Plan
3. NACO User Manual

1.3 Abbreviations and acronyms

The following abbreviations and acronyms are used in this document:

SciOp	Science Operations
ESO	European Southern Observatory
Dec	Declination
eclipse	ESO C Library Image Processing Software Environment
ESO-MIDAS	ESO Image Data Analysis System
FITS	Flexible Image Transport System
IRAF	Image Reduction and Analysis Facility
PAF	PArameter File
RA	Right Ascension
UT	Unit Telecope
VLT	Very Large Telescope

1.4   Stylistic conventions

The following styles are used:

bold   in the text, for commands, etc., as they have to be typed.

italic   for parts that have to be substituted with real content.

box   for buttons to click on.

`teletype`   for examples and filenames with path in the text.

Bold and *italic* are also used to highlight words.

2   Data Reduction Software

2.1   The NACO pipeline

2.1.1   Raison d'etre

The NACO pipeline is an ESO development to provide reduced data to service mode users, visitors at the telescope and operators who execute service mode observations. The pipeline runs in two modes: on-line (for both service and visitor mode programs) and off-line (for service mode programs). On-line data reduction happens in the control-room during the night without any human intervention. The goal here is to provide enough information to assess the quality of the acquired data in the shortest possible time. This quick feedback is often useful e.g. to check the image quality in imaging modes. On-line reduced data are not automatically distributed to observers. Off-line data reduction happens before the data are packed and distributed. The main difference between the on-line and off-line versions is that calibration frames, such as flats and darks, are used. In both cases, it should be understood that the pipeline does not replace careful data analysis and the trained eye of the astronomer. The pipeline implements recipes that are meant to work in most cases. Most astronomers choose to take the pipeline input as a first guess, then refine some points where the default method is not providing the best results for what they are doing.

2.1.2   Downloading and installing

The NACO pipeline is entirely based on the eclipse library. You need to download the library to compile the pipeline. The packages can be found at: http://www.eso.org/projects/aot/eclipse/ Installation instructions are provided with the package and on the Web site. The current version (released Sept. 2005) is 5.0. You always need to download the eclipse-main package, which contains all the libraries and documentation for main commands. After that, you only need to download the package for the instrument or language binding you are interested in (ADONIS, ISAAC, Python, Lua, CONICA, ...). However, some routines for ISAAC are also usable for NACO, so it is better to install both. Please note that NACO is called CONICA... After installation is complete, you should end up with a number of executables in eclipse/bin. The main program you are looking for is called **conicap**, it contains the recipes for most of the NACO data reduction modes, together with some basic documentation for the command usage.

End of June 2007, the NACO pipeline (version 3.6.2) has been publicly released. It can be downloaded from http://www.eso.org/pipelines. This webpage contains detailled information on what to download and how to install the pipeline, as well as an user's manual describing its functionnalities in detail. We advise user's to refer to it for the latest news. The present document remains as information basis for beginners.

2.1.3   Pipeline usage

In eclipse version 4.0 and later versions, most of the NACO pipeline is contained in a single Unix command called **conicap**. Additional commands are **jitter** for imaging jitter, and **spjitter** for spectroscopy. Once you have compiled **conicap**, you can start launching it with no argument to get a list of supported recipes. To get more information about the command itself, try **conicap** man. To get more information about one recipe, try **conicap man recipe**,

where recipe is the name of the recipe you are interested in.

2.2   Data Analysis Tools

You will probably need access to one of the standard data analysis packages for astronomy to help reduce your NACO data: IRAF, MIDAS, IDL, etc. You may want to implement your own routines and scripts to repeat the same algorithm on all your frames, or automate some of the data reduction process. This guide assumes that you have some basic knowledge of such packages.

3   NACO FITS Information

Ancillary data attached to NACO files are all written into FITS headers. This chapter lists the most important keywords and describes commands on how to retrieve them. For ease of reading, keywords are shortened from ESO A B C to A.B.C (shortFITS notation). Note that all of this information is present in the ESO dictionaries, available from the ESO archive Web site. This chapter only tries to summarize the most important information.

3.1   Extracting FITS Information

There are many tools to extract and parse FITS headers. One convenient way of extracting FITS information and displaying it on a terminal or re-directing it to a text file, is to use two stand-alone programs called dfits and fitsort. Both are included into the eclipse distribution.

dfits dumps a FITS header on stdout. You can use it to dump the FITS headers of many files, to allow the parsing of the output. Example: dfits *.fits | grep "TPL ID" Usually, you want to get the value of a list of given FITS keywords in a list of FITS files.

fitsort reads the output from dfits, classifies the keywords into columns, and prints out in a readable format the keyword values and file names. Example: dfits *.fits | fitsort NAXIS1 NAXIS2 BITPIX

fitsort also understands the shortFITS notation, where e.g. ESO TPL ID is shortened to TPL.ID. A classification example could be (both commands are equivalent, since fitsort is case-insensitive):
dfits *.fits | fitsort TPL.ID DPR.TYPE
dfits *.fits | fitsort tpl.id dpr.type

The output from this combination is something like:
```
FILE             TPL.ID                      DPR.TYPE
NACO_0180.fits   NACO_img_obs_GenericOffset  OBJECT
NACO_0182.fits   NACO_img_obs_GenericOffset  OBJECT
NACO_0183.fits   NACO_img_obs_GenericOffset  SKY
NACO_0184.fits   NACO_img_obs_GenericOffset  OBJECT
```
This kind of table is useful in getting an idea of what is present in a directory or list of directories. Loading such a summary table into a spreadsheet program also makes it conveniently readable.

3.2   Visualizing FITS Tables

Similarly, a FITS table can be visualized on the command-line without the need to call a full-fledged FITS-capable package. The dtfits command has been written for precisely that purpose. You will find it useful for spectroscopic data reduction, if you need to check out the results of the pipeline recipes that produce FITS tables. dfits and fitsort will help you classify tables and see ancillary data attached to them, but dtfits will display all information contained in the table itself, in ASCII format on the command-line. There are various options

to help making the output readable on a terminal, or by a spreadsheet program. See the dtfits manual page to get more information.

3.3 File Names

PIPEFILE (if set) contains the name of the product using the official naming scheme for NACO products. This name can be set using the renaming recipe ("rename").
ORIGFILE contains the name of the file on the instrument workstation.
ARCFILE is the archive file name.

3.4 Telescope Keywords

Here is a non-exhaustive list of telescope keywords.
RA Right ascension (J2000) in degrees. Note that the comment field indicates the value in hh:mm:ss.s format.
DEC Declination (J2000) in degrees. Note that the comment field indicates the value in hh:mm:ss.s format.
ADA.POSANG Position angle on sky as measured from North to East.(degrees).
TEL.AIRM.START Airmass at start.
TEL.AIRM.END Airmass at end.
TEL.AMBI.FWHM.START Astronomical Site Monitor seeing at start. Note that this value might differ significantly from the NACO image quality, which is usually better.
TEL.AMBI.FWHM.END Astronomical Site Monitor seeing at start. Note that this value might differ significantly from the NACO image quality, which is usually better

3.5 Instrument Keywords

Here is a non-exhaustive list of instrument keywords.

INS.MODE	=	instrument mode. It is a label codifying the mode (imaging, spectroscopy, etc.) used for the current frame. See the NACO User's Manual for more information about possible values of this keyword.
INS.OPTI5.NAME, INS.OPTI6.NAME	=	Name of the filter. Note that the J band filter is placed in INS.OPTI4.NAME
INS.OPTI1.NAME	=	Wheel with the entrance slits & masks
INS.OPTI4.NAME	=	Wheel containing the polarising elements and grisms
INS.CWLEN	=	Central wavelength (microns). Note that this value is a rough estimate.

INS.FPI.LAMBDA	=	FPI Wavelength (nm)
INS.FPI.N	=	FPI order
INS.PIXSCALE	=	Pixel scale.
INS.OPTI7.NAME	=	Name of the objective.

INS.LAMP1.ST	=	Argon lamp status.
INS.LAMP2.ST	=	Halogene lamp status.
DET.DIT	=	Detector Integration Time (seconds)
DET.NDIT	=	Number of averaged DITs.
DET.MODE.NAME	=	Detector readout mode.
INS.PIXSCALE	=	Pixel scale in arcseconds per pixel.

3.6 Adaptive Optics related Keywords

A number of keywords are specific to the adaptive optics system, and are present only in files actually using the AO system. Here comes the list of Real Time Computer (RTC) related keywords.

AOS RTC DET WFS MODE	=	Wavefront sensor (WFS) mode
AOS RTC DET DST L0MEAN	=	Average value of L0 (calculated by RTC on real data), where L0 represents the outer scale
AOS RTC DET DST R0MEAN	=	Average value of R0 (calculated by RTC on real data), where R0 represents the Fried's parameter
AOS RTC DET DST T0MEAN	=	Average value of T0 (calculated by RTC on real data), where T0 represents the coherence time of the atmosphere
AOS RTC DET DST NOISMEAN	=	Average value of noise of variance (calculated by RTC on real data)
AOS RTC DET DST ECMEAN	=	Average value of coherent energy (calculated by RTC on real data)
AOS RTC DET DST FLUXMEAN	=	Average value of reference source (calculated by RTC on real data)
AOS RTC DET ADAO ST	=	AO correction activated

3.6.1 Adaptive Optics System specific

| AOS INS DABE ST | = | Correction of static aberrations flag |
| AOS INS WABE ST | = | Correction of static aberrations flag |

| AOS INS REFR ST | = | Differential refraction compensation (VISIBLE WFS) |
| AOS INS TRAK ST | = | Tracking of the reference object (for differential tracking) |

AOS INS DICH POSNAM	=	Dichroic name position
AOS INS WSEL POSNAM	=	WFS selector named position
AOS INS MFLX ST	=	Correction of mechanical flexure
AOS OCS WFS TYPE	=	Wavefront sensor type
AOS OCS WFS MODE	=	Wavefront sensor mode
OCS SOS INS MODE	=	Instrument mode used

3.6.2   Infrared wavefront sensor

AOS IR DET WFS MODE   =   Wavefront sensor mode
AOS IR INS FILT POSNAM   =   Filter used with IR WFS

3.7   Observation Block Keywords

OBS.PROG.ID   =   Program ID.
OBS.NAME   =   Name of the OB (as prepared with P2PP).
OBS.TARG.NAME   =   Target package name (as prepared with P2PP).

3.8   Template Keywords

3.8.1   Keywords common to all templates

TPL.ID contains an unique identifier describing the template which was used to produce the data. Frame selection in the pipeline is mostly based on this keyword value.
DPR.CATG Data Product category (SCIENCE, CALIB, ...).
DPR.TYPE Data Product type (OBJECT, SKY, ...).
DPR.TECH Data Product acquisition technique (e.g. IMAGE, SPECTRUM).
TPL.NEXP Number of scheduled exposures within the template.
TPL.EXPNO Exposure number within template.
A template may produce several different frame types. Frames are discriminated by the value of the DPR keywords: DPR.CATG, DPR.TYPE, and DPR.TECH take different values depending on the observed frame type.

3.8.2   Keywords common to the jitter templates

The offsets sent to the telescope for jitter observations, both in imaging and spectroscopy, are stored into 8 keywords. This applies to AutoJitter, AutoJitterOffset, and GenericOffset templates.
SEQ.CUMOFFSETX and SEQ.CUMOFFSETY for cumulative offsets in pixels.
SEQ.CUMOFFSETA and SEQ.CUMOFFSETD for cumulative offsets in arcseconds (alpha, delta).
SEQ.RELOFFSETX and SEQ.RELOFFSETY for relative offsets in pixels.
SEQ.RELOFFSETA and SEQ.RELOFFSETD for relative offsets in arcseconds (alpha, delta).

Cumulative offsets are always relative to the first frame in the batch (TPL.EXPNO=1). Relative offsets are always relative to the previous frame (TPL.EXPNO-1) in the batch. If the same guide star is used before and after an offset, the offsetting accuracy is about 0.1 arc seconds. All recipes looking for offset information take this into account and will use the header offset information as a first guess and will refine the offset through cross-correlation techniques. In AutoJitter mode, the jitter offsets are generated using a Poisson distribution. SEQ.POISSON is an integer describing the Poisson homogeneity factor used for this distribution. See the eclipse web page (http://www.eso.org/eclipse) for more information about this factor.

The jitter recipe from eclipse always expects offsets to be given in pixels, not in arcseconds. If your headers do not mention the offsets in pixels, you must translate arcseconds to pixels

yourself and feed the information back into the jitter command. The input offsets are then given by an ASCII file instead of being read from the FITS headers.

3.9   Pipeline Products

To allow identification of pipeline products, some keywords are inserted in the output FITS headers.

PIPEFILE is a standard 8-char FITS keyword which contains the name of the file as set by the pipeline when creating the product, it is useful as a label to identify the file. Nothing requires this name to be set, it is only here for convenience.

PRO.DID contains the version number of the dictionary used for all product keywords.

PRO.TYPE contains the type of the data products as one of TEMPORARY, PREPRO-CESSED, REDUCED, or QCPARAM.

PRO.CATG is probably the most important product keyword, since it labels each frame with a product ID unique to the recipe. It qualifies files with hopefully understandable product labels, e.g. `NACO_IMG_DARK_AVG`.

PRO.REC1.ID identifies the recipe that generated the file, with a unique name.

PRO.REC1.DRS.ID identifies the Data Reduction System that was used to produce the file.

PRO.DATANCOM specifies the number of raw frames that were combined to generate the product. Its exact meaning depends on the recipe, see recipe documentation to learn what it refers to.

3.10   Quality Control Products

Quality control (QC) outputs QC parameters in the fits headers of the reduction product, e.g. `QC.SLIT.POSANG` for the slit position angle).

3.11   Useful commands to browse the FITS headers

Exemple of useful commands to browse fits headers:

Imaging: dfits *.fits | fitsort OBJECT DET.DIT DET.NDIT TPL.NEXP DET.NCORRS.NAME DET.MODE.NAME INS.OPTI7.NAME INS.OPTI6.NAME INS.OPTI4.NAME INS.OPTI5.NAME INS.MODE | grep IMAGING

Spectroscopy: dfits *.fits | fitsort OBJECT EXPTIME AIRMASS INS.OPTI7.NAME INS.OPTI4.NAME INS.OPTI6.NAME DET.EXP.NAME DET.NCORRS.NAME DET.MODE.NAME

Astrometry: dfits *.fits | fitsort OBJECT CRPIX1 CRPIX2 CRVAL1 CRVAL2 CDELT1 CDELT2 CD1_1 CD2_2 CD2_1 CD2_2 INS.PIXSCALE

Adaptive Optics configuration: dfits *.fits | fitsort OBJECT AIRMASS AOS.OCS.WFS.TYPE AOS.INS.DICH.POSNAM AOS.OCS.WFS.MODE AOS.RTC.DET.DST.R0MEAN TEL.AMBI.FWHM.START AOS.RTC.DET.DST.ECMEAN

Flat field: dfits *.fits | fitsort TPL.NAME DET.NCORRS.NAME DET.MODE.NAME INS.OPTI7.NAME INS.OPTI6.NAME INS.OPTI4.NAME INS.OPTI5.NAME INS.LAMP2.NAME INS.LAMP2.SET

Detector configuration: dfits *.fits | fitsort OBJECT DET.NCORRS.NAME DET.MODE.NAME DET.DIT DET.NDIT TPL.NEXP

4   Frames needed for the reduction

4.1   Darks

Dark frames are exposures without detector illumination. The dark current of the detector is small, so the dominant feature in these frames is the detector bias, which is also called the zero level offset, since it is not possible to take a zero second exposure with the array. Usually one takes at least three darks and combines them with suitable clipping to create the dark frame that is subtracted from the science data. As the bias is a function of the DIT, the DIT of the science data and that of the dark must match. Dark frames are acquired through a dedicated template, which obtains (usually at the end of the night) at least three dark frames for each DIT and detector readout mode that was used during the night. The darks can be reduced with the dark recipe: **naco_img_dark**. This recipe will produce one dark frame for each DIT. The readout noise is also measured from these frames and the information is written in the fits header of the result file. The dark recipe is actually separated into two sub-recipes, the first one creates the dark and the second one computes the readout noise. Each sub-recipe can be called individually by using the dark-avg and dark-ron recipes. Currently, the dark created by dark-avg is nothing more than an average of the input files (after having them sorted by identical DIT and detector mode).

In NACO due to a poor shielding of the detector, darks (longer than 30sec) taken with different cameras show different structures. As a consequence darks are also taken with the same camera as the one used during the night for the science observations. The pipeline takes this into account to sort out the data.

4.2   Twilight and sky Flats

Imaging data can be flat fielded with sky flats. The flats are derived by imaging a region of the sky relatively free of stars. Between 5 to 10 exposures with constant DIT and NDIT are taken for each filter and a robust linear fit between the flux in each pixel and the median flux of all pixels is used to produce the flat field.

Pipeline implementation: for each pixel on the detector, a curve is plotted of the median plane value against the individual pixel value in this plane. This curve shows the pixel response from which a robust linear regression provides a pixel gain. The image showing all pixel gains (i.e. the flat-field) is normalized to have an average value of 1. By-products of this routine are: a map of the zero-intercepts and an error map of the fit. Both can be used to visually check the validity of the fit. A bad pixel map can also be produced by declaring all pixels whose value after normalization is below 0.5 or above 2.0 as bad. The eclipse recipe is called **naco_img_twflat**.

For the specific case of sky flats taken with the filters L and M, the sky is relatively insensitive to changes in the twilight sky, so one cannot use the method used for SW data. Instead, sky flats are taken at 3 different airmass (1., 1.5, 2.0) with the same exposure time. The difference in airmass turns into flux level in the image. The flat can then be created by subtracting the images that were taken at one of the higher airmasses from the image taken at Zenith. The resulting image is then normalised to 1.

4.3   Lamp Flats

Every morning, after the end of the night and with the telescope closed, (halogen) lamp flats are taken in each combination of filter, camera and readout mode of the detector used during the night. These flats are internal to CONICA and do not include any effects coming from the telescope or NAOS. It is actually impossible to take sky flats for all combinations. These lamp flats are taken by pairs, lamp on and off.

Spectroscopic lamp flats are also taken in the same way.

Pipeline implementation: for each pair of image, the routine produces a normalized image with median to 1. ADU. The eclipse recipe is called **naco_img_lampflat**. The resulting flat field is similar to a sky flat within 5%.

4.4   Lamp arcs

Every morning, after the end of the night and with the telescope closed, spectroscopic lamp arcs are taken in each spectroscopic setup used during the night. These arcs are internal to CONICA and taken with an Argon lamp. These lamp flats are taken by pairs, lamp on and off. The Argon lamp does not produce any arc in wavelength longer than $2.5\mu m$ so no arcs are available in L and M bands.

Pipeline implementation: the routine detects vertical arcs in a spectral image, models the corresponding deformation (in the x direction only) and corrects this deformation. Finally a wavelength calibration using a lamp spectrum catalog is made using vertical arcs. For each pair of image, the routine produces a table calibrated in wavelength. There is no specific recipe to reduce NACO arcs but one can use the ISAAC recipe (**isaac_spc_arc**) but be careful that the NACO arcs are orthogonal to the ISAAC ones.

5   Zero Points

Standard stars are observed every night in the J, H and Ks filters, S27 objective, visible dichroic. For other combinations of filters, objective and dichroics, standards are observed as required.

Standard stars are imaged over a grid of five positions, one just above the center of the array and one in each quadrant. The recipe finds the standard (it assumes that the star in the first image is near the center), computes the instrumental magnitude, and then uses the standard star database to determine the ZP, which is uncorrected for extinction.

The standard star database contains about 1000 stars with magnitudes in the J, H, K, Ks, L and M bands, although most stars only have magnitudes in a subset of these filters. Stars are currently taken from the following catalogs: Arnica, ESO, Van der Bliek, LCO Palomar, LCO Palomar NICMOS red stars, MSSSO Photometric, MSSSO Spectroscopic, SAAO Carter, UKIRT extended, UKIRT fundamental.

The implemented recipe is the following: For any couple of consecutive images (image1, image2):

1/ diff = image1 - image2

2/ Locate in diff the star around the expected pixel position (provided by the FITS header or by an external offset list).

3/ Compute the background around the star, and the star flux.

4/ Store the flux result in an output table.

Apply steps 2 to 4 to the inverted image image2-image1. This yields 2(N-1) measurements for N input frames. From this statistical set, the highest and lowest values are removed, then an average and standard deviation are computed.

The conversion formula from ADUs to magnitudes is: zmag = mag + 2.5 * log10(flux) - 2.5 * log10(DIT) where: zmag is the computed zero-point. mag is the known magnitude of the standard star in the observed band. flux is the measured flux in ADUs in the image. DIT is the detector integration time.

Note that neither the extinction nor the colour correction are included in the ZP pipeline calculation. The average airmass is given in the output result file, together with individual airmass values for each frame. The average extinction on Paranal for the J, H, Ks and NB M filters, is available from the ISAAC web pages.

QC parameters contains two filters: QC.FILTER.OBS, indicating the band at which the observations have been performed, e.g. K_s and QC.FILTER.REF, which is the closest filter given in the catalog and matching the observation, e.g. K. This correspondence completely ignores corrections due to filter mismatch, and, in some cases, these corrections are substantial. It is possible to use the **naco_img_zpoint** recipe to compute the zero point.

6   Imaging without chopping

This section deals with imaging at all wavelength, with broad or narrow band filters, without chopping. Fabry Perot imaging, SDI and coronography are somewhat specific and are treated separately.

Remember that using an AO system, images suffer from anisoplanatism, -the field dependence of the PSF-. It corresponds to the angular decorrelation of the wavefront coming from two angularly separated stars. This phenomenon affects the quality of the AO correction in the direction of the target when the reference star is not on axis, but it can also affect other parts of the field, this depends on the conditions at the time of the observation.

6.1   Cosmetics and vignetted regions

The CONICA detector cosmetic is very good. If one wants the very best cosmetic, then the FowlerNSamp detector mode has to be chosen. But even with the other readout modes, the bias structures will easily be canceled by a sky subtraction.

Occasionally, an electronic noise would affect the images, it is due to a bad shielding and occurs in a few positions of the telescope rotator, and can disappear changing the rotator angle, but there is no routine for a posteriori filtering.

Using the S54 objective, there is a small vignetting on the left and bottom of the images, this is due to a misalignment of the internal mask and can not be corrected.

Dead and hot pixels are present all over the detector, and depend on wavelength and readout mode. Some bad pixels maps can be created by the routine used to reduce dark frames (Sect. 4.1). An alternative is to use sigma-clipping filtering.

6.2   Dark subtraction

In the morning following the observations, dark frames are taken for each detector setting used during the night. These darks are archived and can be used for subraction to the science images, although most of the time it will not be necessary since sky frames will be prefered.

6.3   Flat fielding

Before any other operation, one divides all images with a flat field which has been normalised to unity. With NACO, flat fields are taken using a lamp flat in the morning, and on sky in the evening before sunset. Both are delivered to the users. A more detailed discussion on how the flat field can be created is given in a Sect. 4.2 & 4.3).

6.4   Sky subtraction

This is the most important step and great care and a good understanding of the technique are necessary if good results are required. This is particularly important for deep imaging as an error at the 0.01% level will significantly effect the photometry of the faintest sources. There are two cases. In relatively blank fields, the sky is created from the object frames themselves. For crowded fields or large objects, the sky is created from frames that were specifically taken

to measure the sky. For deep exposures, the sky is computed from a subset of exposures and there will be one sky frame for each object frame. For accurate photometry, it is very important that the object frame is not included in the frames that are used to compute the sky. This is a weakness of the current jitter recipe. For H and K band observations the sky frame should be computed from frames that were taken 5-10 minutes either side of the object frame. For J band observations, these numbers can be doubled. For conditions where the sky background is varying rapidly (clouds or observations taken just after evening twilight) a more rapid sampling of the sky is necessary. All sky frames contain objects, so one has to combine them with suitable clipping. A robust method is to first scale frames to a common median and then remove objects by rejecting the highest and lowest pixels. Rejecting the two highest and two lowest pixels would produce even more robust results. The remaining pixels are then averaged (the median can also be used, but it is a noisier statistic). The resulting sky frame is then scaled (to match the median of the object frame) and subtracted.

A more sophisticated approach is to do the sky subtraction in two steps. The first step reduces the data as described above, produces the combined image and then locates all the objects. In the second step, the data is rereduced with the knowledge of where the objects are. These objects are then excluded when the sky is estimated in the second pass. This is the approach used by the XDIMSUM package in IRAF and for very deep imaging it is the recommended package.

Use the **naco_img_jitter** recipe to reduce your imaging data.

6.5   Image registration and stacking

To register the sky-subtracted images to a common reference, it is necessary to precisely estimate the offsets between them. Jitter applies a 2d cross-correlation routine to determine the offsets to an accuracy of 1/10th of a pixel. There are other ways to find out offsets between frames: with many point-sources, point-pattern matching is a possibility. Identifying the same objects in all consecutive frames would also yield a list of offsets between frames.

An initial estimate of the offsets between frames can be found in the FITS headers. jitter assumes that the offsets found in the input FITS headers have a certain accuracy. If there are no input offsets, they are all initially estimated to be zero in both directions.

Registering the images is done by resampling them with subpixel shifts to align them all to a common reference (usually the first frame). Resampling can make use of any interpolation algorithm, but be aware that using cheap and dirty algorithms like nearest-neighbor or linear interpolation can degrade the images by introducing aliasing. jitter offers many higher-order interpolation kernels that introduce few or no artifacts; however, the noise (high frequencies) will be smoothed a little bit.

Stacking the resulting images is done using a 3d filter to remove outliers and jitter gives you a choice between 3 different filters. Linear means that all frames are actually averaged without filtering (pass-all filter). This is not recommended as this is likely to keep cosmic rays and other outliers in the final frame. Median means that the final frame is the median of all resampled frames. The last filter (default) scales all frames by their medians and removes the highest and lowest pixel values before taking an average.

Note that in the current version of the pipeline jitter recipe, the final frame is an union of all input images (as opposed to an intersection for previous versions), which means that it is bigger than any of the initial input frames.

Use the **naco_img_jitter** recipe to reduce your imaging data.

6.6   A few words about the Eclipse recipe jitter to reduce your imaging data

jitter reduces images taken in infrared jitter imaging mode. It makes a number of assumptions on the input signal and has a list of several possible algorithms with associated parameters for each reduction stage.

jitter has been developped to reduce jitter imaging data taken from infrared instruments, e.g. IRAC2, SOFI, ISAAC or NACO. Although some features are specific to the two latter instruments, it is reasonable to think that the same algorithms should work on similar data.

jitter is configured through an initialization file. The name of this file is defaulted to jitter.ini but can be changed through the use of the -f option. In the following documentation, this file is referred to as the ini file.

The jitter data reduction process is divided into flat-fielding/dark subtraction/bad pixel correction, sky estimation and subtraction, frame offset detection, frame re-centering, frame stacking to a single frame, and optional post-processing tasks. Some processes may be deactivated depending on what you intend to do with the data. Describing all the algorithms in this command is far beyond the scope of this manual page. Please refer to the pipeline user's manual for more details.

To setup the process, you need first to generate a default jitter.ini file and then change parameters according to your needs. This initialization file is self-documented.

7   Coronagraphic Imaging

7.1   Cosmetics and vignetted regions

Idem as Imaging without chopping.

7.2   Dark subtraction

Idem as Imaging without chopping.

7.3   Flat fielding

For classical Lyot coronography using the ($\oslash = 0.7$ and $1.4\,''$) occulting masks, the same strategy as Imaging without chopping can be followed to properly flat field the sky and science frames.

For coronography using the semi-transparent and 4QPM masks, the support of these 2 optical masks may move for different telescope positions during the night. Therefore, it is clearly recommended to observe night time flat fields just after the science observations. A dedicated template for night time calibration has been created and these flats should be used to reduce the science and sky coronagraphic observations in the same way as documented before.

7.4   Sky subtraction

As for Imaging without chopping, great care is necessary to properly subtract the sky from the science coronographic images. For all coronagraphic templates, the sky is created from frames that were specifically taken to measure the sky (frame in open loop distributed randomly around the object position at minimum distance of $> 30\,''$). Classically, the sky is computed from a subset of frames and exposures (i.e NDIT$\times$NEXPO) similar for the science and the reference star. The integration time must be equal. The total observing time spent on the sky can be reduced, however one should be aware that the sensitivity in the sky-subtracted coronographic image will be limited by the sky and not the science frame.

7.5   Image stacking

When using the coronographic templates, one can decide to move the telescope alternatively between a fixed object position and a sky position, i.e to repeat several times a number of AB cycle between the object and the sky. By repeating such sequence, the bright source position behind the mask will not be kept exactly at the same position. Therefore, to stack together all sky-subtracted coronographic images of each AB cycle, the images should before be properly shifted and added, either using faint sources if present in the science image or the central source PSF wing. No ESO tool are provided for this reduction/analysis step.

7.6   Central star subtraction

In the case of coronographic observations, one is generally interested by the faint environment of the bright source to look for faint extended structures (circumstellar or galactic disks) or

faint stellar and/or sub-stellar companions. A main step in data reduction is the removal of the scattered light remaining around the mask. This implies to use a reference single star whose scattered light is scaled to that of the object of interest. The scaling factor is generally estimated by azimuthally averaging the division of the star of interest by the reference star. In the case of NACO, the instrument is mounted at the Nasmyth B focus of an Alt/Azimuthal telescope. For this reason the pupil is rotating while the FoV is kept fixed. A critical consequence is that the reference star must be chosen with great care to keep the same pupil configuration (i.e the same parallactic angle) for the science and the reference star observations. If not, the telescope aberrations will not be the same. This will then cause important residuals in the science image subtracted from the re-centered and scaled reference star. No ESO tool are provided for this reduction/analysis step or for the reference star selection.

7.7   Coronography dedicated to astrometry

In the case of coronographic observations dedicated to relative astrometric measurements between the bright source and faint companion(s), a special template has been written.

8   Simultaneous Differential Imaging

The first steps of data reduction are related to the cosmetics, dark subtraction and flat fielding. Information provided for imaging without chopping apply here as well.

The following steps of data reduction greatly depends on how data have been taken, e.g. have sky frames been observed?; but also and essentially on the kind of science one aims at. For detailed information we refer to three papers written by the consortium, who built the NACO SDI:

- Biller B. et al, 2004, SPIE 5490, 389–397:
  Suppressing speckle noise for simultaneous differential extrasolar planet imaging (SDI) at the VLT and MMT
  This paper presents the genenralities of the SDI data reduction.

- Biller B. et al., 2006, Proc. IAU Colloquium #200, Aime C. & Vakili F., 571–576:
  Suppressing speckle noise for simultaneous differential extrasolar planet imaging (SDI) at the VLT and MMT
  This paper gives a block diagram explaining a kind of pipeline reduction of the data.

- Biller B. et al, 2006, SPIE 6272, 62722D-1–62722D-10:
  Contrast limits with the simultaneous differential extrasolar planet imager (SDI) at the VLT and MMT
  This paper contains also the 'pipeline block diagram' but also shows the limit of the method.

These papers can be found on the NACO webpages: http://www.eso.org/instruments/naco/tools.

9   Spectroscopy

The most basic way of taking IR spectra is to observe the target along two slit positions. The sky is then removed by a process which is sometimes called double sky subtraction. The basic steps of how to reduce these type of data are detailed hereafter. Note that there is no chopping in spectroscopy. The Prism could be reduced as other settings, but remember that there are no arcs in that case. There is no specific recipe for NACO spectroscopic data reduction in eclipse. One can use instead the ISAAC recipe **isaac_spc_jitter**.

One should remember that reducing spectroscopic data is difficult, and the pipeline may be not accurate enough to produce final data ready for interpretation... So don't use pipeline reduced data to send a paper to Nature claiming the detection of a QSO at $z = 69$ before you are sure of what you actually did !

9.1   Flat fielding

As in imaging one divides by the flat field. This operation must be executed before any other operation, since it affects all the frames.

9.2   Sky subtraction

There are two techniques observers use in taking IR spectra. There is the classic sequence where one observes the object at two slit positions. i.e. ABBA, etc, and there is the more complex case where one has observed the target along several slit positions. In the classical case, one simply subtracts frames taken at different slit positions. So one needs to form A-B, B-A, etc. This simple step removes the bias and results in an image with two spectra, one positive and one negative. In the more complex case, one could build a sky frame from several spectra, as one does when building the sky frame in imaging. This results in an image with only one positive spectrum.

9.3   Slit curvature correction and wavelength calibration

Usually the spectra are strongly curved and tilted. Before the 2d spectra are combined, they need to be straightened. It is useful to do the wavelength calibration at the same time, so that the horizontal axis is in wavelength units. The wavelength scale can be calibrated with either arc frames or the OH lines that are imprinted on each spectrum. The advantage of the arcs is that there are lines covering the entire 0.9 to 2.5 μm range. The disadvantage is that the arcs are taken separately and, in most cases, it means the optical alignment might have changed slightly (e.g. the slit position angle) between the time the target was observed and the arcs were taken. One can use the OH lines to cross check and correct the zero point of the wavelength calibration, which will be a necessary step in most cases. The advantage of the OH lines is that they are numerous and that they lead to a slightly more accurate wavelength calibration. The disadvantages are that: in some regions, particularly beyond 2.2 μm there may be too few lines to do a good fit; in standard star observations, where exposure times are short, the OH lines may be too faint; and when the spectral resolution is low, the OH lines may be heavily blended. For both arcs and OH lines a 3rd order Legendre (4 terms) gives a

good description of the dispersion.

9.4   Combining 2d spectra

For the classical ABBA technique, one multiplies each image by -1 and adds it back to itself after a suitable shift. This method of combining data is often called double sky subtraction as it effectively removes any residual sky that remains after the first subtraction. It results in an image that has one positive spectrum and two negative spectra either side of the positive spectrum. In the more complex cases, one combines the individual spectra after suitable shifts have been applied.

9.5   Extraction

For the classical ABBA technique, one should extract the spectrum without fitting the sky. Fitting the sky only adds noise. For more complex cases, a low order fit to the sky may be required.

9.6   Removing telluric lines

This is a critical step that requires some care and, possibly, some experimentation. The aim is to remove the large number of telluric lines that appear in IR spectra. This is done by dividing the object spectrum with that of a telluric standard, corrected by its own temperature profile (one can use the spectral type, see next section). Since this is a division of one spectrum by another, it is important that the strength, shape and centroid of the telluric lines match.

First and foremost, the telluric standard and the object have to be observed with the same instrument setup, with roughly the same airmass and, if possible, consecutively.

Secondly, the object and science data should be reduced in the same way and with the same calibration frames. For the best results, one may have to modify the spectrum of the telluric standard so that the center and strength of the telluric lines match those of the object spectrum.

The next step is to remove spectral features that have been imprinted onto the object spectrum from the telluric standard itself. Telluric standards are either hot stars or solar type dwarfs. Both types contain spectral features that should be removed. For solar type stars, one can use the observed solar spectrum to remove the features. This can be tricky if the spectral resolution of the instrument is a function of the wavelength as it means that the kernel for convolution also has to be a function of wavelength. The arc spectra and the OH lines can be used to estimate what this function is.

Hot stars usually contain helium and hydrogen lines. If the spectral regions around these lines are of interest, then one should think carefully about using these type of stars. If the resolution is high enough, which is certainly the case for MR observations, one can try to remove these lines by fitting them. Alternatively, one can use a second telluric standard that does not have helium or hydrogen lines so that these lines can be removed from the hot star.

The NACO FITS header does not contain the full target name of the telluric standard, but it is usually reported into the night report. However, sometimes, operators forget to include it in the night report. To find out which telluric standard was used, look at the RA and DEC of the target and consult the list of Hipparcos stars that are often used as telluric standards.

9.7   Flux calibration

The first step is to obtain a relative flux calibration. The second step is to do absolute flux calibration. If the telluric standard was a hot star, then a blackbody curve can be used to model the continuum of the standard. The spectral type of the star can be used to give an idea of what temperature to use. The blackbody curve is then multiplied into the object spectrum. For solar type stars, a blackbody curve is a good enough description of the spectral energy distribution above $1.6\mu m$. Below $1.6\mu m$, a more accurate description of the continuum is required. The spectral energy distribution of a wide variety of stars are available through the ISAAC web pages.

The second step is absolute flux calibration. If the magnitude of the target is known, a rough calibration can be obtained by convolving the spectrum with the filter curve and determining the appropriate scaling. Otherwise, determining the absolute flux calibration is more difficult and less certain. Anyway, due to the slit (very narrow) size, it is very hard to get an absolute flux calibration with NACO.

9.8   How to use isaac_spc_jitter

The pipeline recipe that reduces SW spectroscopic data is called isaac_spc_jitter.

The recipe uses an initialisation file, which contains parameters that define how the recipe runs. The "-g" option allows to create an initialisation file called spjitter.ini. This file can be edited and modified by the user to setup some parameters. The initialisation file contains parameters which indicate which table to use to correct for slit curvature which table to use for wavelength calibration which table to use to correct for spectral tilt (the one produced by the startrace recipe) and which flat field to use The input file should be an ASCII file listing the raw FITS files.

The recipe starts by classifying the input images according to the cumulative offsets in the headers. The classic way of taking IR spectroscopic data is to observe the target along two positions along the slit, which we will call A and B. An example, may be the sequence AAABBBBBBAAA. After flat fielding all the data, the recipe will take the first three A frames and average them, take the first three B frames and average them, etc. The recipe then subtracts one average from the other, corrects for slit curvature, spectral tilt and wavelength calibrates. If the tables for the wavelength calibration and the correction for slit curvature are missing, the recipe will use the OH night sky lines. If there are too few of these, the recipe will use a model to do the wavelength calibration and will skip the correction for slit curvature. If the flat field is missing, the recipe skips the flat fielding step.

The subtracted frames will contain positive and negative spectra. The two spectra are combined by multiplying the image by -1 and adding it to the original after a suitable shift. The resulting frames are then added together to give the final result.

At the end, a spectrum can be extracted. Either the user specifies the position of the spectrum they want to extract in the initialisation file, or the spectrum of the brightest object is extracted.

Pease refer to the ISAAC data reduction cookbook for details.

10   Imaging with chopping

The M band filter imaging can only be done using chopping and hardware windowing of the detector. This is because the thermal background is so high that it would saturate the detector in less than a fraction of a second. The technique of chopping involves rapid sampling of the sky by moving the secondary mirror in phase with the read out of the detector. For NACO, the typical distance between the two positions (the chop throw) is 10 arcsec and the chopping frequency is typically around 0.1 Hz.

In what follows, the ON beam refers to the positive image, the OFF beam refers to the negative image. In addition to chopping the telescope nods so the ON beam at position A overlaps with the OFF beam of position B. The typical nodding sequence is ABBAABBA ... etc.

To allow AO correction in all positions, a dedicated mirror, called field selector mirror, is used to provide counter-chopping and keep the reference star still into the WFS beam. Thus, the AO loop is always closed in all positions, chopping and nodding. All the system is sinchronized so that the science frame integration starts only when the AO loop is closed. This mode produces huge overheads.

For the data taken with chopping, one gets two sorts of frames: chopped frames and half cycle frames. Chopped frames correspond to the difference between frames taken with the secondary at the two positions (i.e. ON and OFF beams). In these frames, the sky has already been subtracted and one has both positive and negative images. This is only for the real time display. Only the half cycle frames are saved to disk and sent to the archive. These frames are stored in a cube. The first plane in the cube corresponds to the ON image and the second plane corresponds to the OFF image, and so on. In these frames, the sky has not been subtracted and images are always positive.

10.1   Cosmetics

Using the Uncorrelated readout mode at extremely fast rate, many hot pixels appear. This is a real concern. As usual, maps of bad pixels are produced by the **naco_img_dark** recipe.

10.2   Sky flat

As usual, images must be divided by a sky flat.

10.3   AB Subtraction chop Registration and stacking

Images come by pairs (half cycle frames), on each chop position, to be subracted one from the other. Since chopping comes with nodding, the end result will be a central positive image with an exposure scaling of 2*DIT seconds, and two negative images with an exposure scaling of DIT seconds.

10.4   using the jitter recipe

The recipe reduces data by:
- classifying the input frames into those taken with the secondary at either side of the nod, i.e. positions A and B.
- computing (A-B)/2. Each frame shows one positive and two negative objects.

computing the offsets between frames. Initial estimates are taken from the FITS headers and further refined through cross-correlation.

- registering the frames and stack into a single image.

11   Additional useful Recipes

There are some other eclispe commands very useful to perform basic operations on images. Since they are unix command lines, they are very fast.

Eclipse command	Definition	Pipeline recipe where applicable
average	average (or median average) list of frames or a cube to a single frame	
catcube	concatenate data cubes or create a data cube from a list of frames	
ccube	cube computer	
collapse	image collapse along X or Y	
deadpix	bad pixel map handling	
dfits	read keywords form FITS header	
distortion	Distortion estimation routine (spectroscopy)	
dtfits	FITS table dump	
dumppix	dump pixels to stdout	
encircl	radius for given percentage encircled energy	
extract	extract data from a cube	
extract_spec	Spectrum extraction	
fitsort	read specific keywords form FITS header (with dfits)	
flat	create linear gain maps out of twilight data cubes	naco_img_twflat
jitter	jitter imaging data reduction	naco_img_jitter
peak	object detection and stat computation	
spjitter	spectroscopic jitter data reduction	isaac_spc_jitter
stcube	cube statistics	
strehl	Strehl ratio computation	naco_img_strehl

All these eclipse commands have a man menu with more detailed information.

$$_{--}\mathbf{oOo}_{--}$$