• U C •

FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

João Guilherme Basílio dos Santos [1,2]

Photometry Data Processing for ESA's CHEOPS Space Mission

Thesis submitted to the
University of Coimbra for the degree of
Master's in Astrophysics and Space Instrumentation

Supervisors:
Prof. Dr. Sérgio Sousa[1]
Prof. Dr. João Fernandes[2]

Coimbra, 2018

[1] Center of Astrophysics, University of Porto, Rua das Estrelas, 4150-762 Porto, Portugal
[2] Physics Department, University of Coimbra, Rua Larga, 3004-516, Coimbra, Portugal

This work was developped in collaboration with:

Center for Astrophysics of the University of Porto

Centro de Astrofísica
da Universidade do Porto

Acknowledgments

First, I want to thank my two supervisors in Porto, Dr. Sérgio Sousa and Dr. Olivier Demangeon, for the help and guidance given throughout the academic year. A special thanks to Dr. Sérgio Sousa for allowing me to do this project in CAUP, and to CAUP for giving me the necessary conditions to do my thesis. I would like to also give a special thanks to Dr. Olivier Demangeon for the valuable inputs given on my thesis work and for the important discussions in the later stages of this thesis.

I want to also thank my family, in particular my parents, for always helping me in each way they could .

I would like to thank my friends, the old and the new, and my praxe family ”Os Fonitos” for making some memorable academic nights.

Finally, I want to thank my girlfriend, Laetitia, for coming to live in Porto. You made my days brighter.

Resumo

Este trabalho tem por objectivo estudar e melhorar a *pipeline* de redução de dados da missão CHEOPS. Esta pipeline é um software cuja finalidade é a correcção e redução dos dados recolhidos por esta missão espacial. Neste trabalho apresentamos uma descrição da estrutura e dos módulos principais da pipeline, dando destaque ao módulo da calibração. A pipeline é depois replicada, de modo a facilitar o desenvolvimento de algoritmos adicionais e o seu consequente teste. É também efectuado o desenvolvimento e teste de um algoritmo melhorado da correcção de não-linearidade do detector. Os procedimentos de teste e os respectivos resultados são também apresentados e discutidos.

Abstract

This work aims at studying and improving the CHEOPS mission data reduction pipeline. This pipeline is a software whose objective is to correct and reduce the RAW data that will be collected throughout the course of the mission. We present a description of the structure and the main modules of the pipeline, with a focus on the calibration module. The pipeline is then replicated, to make the development of additional algorithms and their testing easier. In addition, we develop and test an improved algorithm for the non-linearity corrections of the CHEOPS CCD. The testing procedures and its results are also shown and discussed in this thesis.

Glossary and Acronyms

Glossary

Imagette A 30*30 sub-array centered on the target.

Individual/Unstacked Image An image which hasn't been stacked.

Observation A set of exposures.

Original Uniform Correction Method The first implemented method used in the pipeline based on the average flux from multiple stacked images. Also known as the splines method.

Acronyms

ADC Analog-to-Digital Converter.

ADU Analogic-to-Digital Units.

CCCD CHEOPS CCD.

CCD Charged Coupled Device.

CHEOPS Characterising Exoplanet Satellite.

CR Cosmic Ray.

CTE Charge Transfer Efficiency.

DN Digital Number.

DRP Data Reduction Pipeline.

ESA European Space Agency.

FF Flat-Field.

LOS Line-of-Sight.

P2P Point-to-Point Precision.

ppm parts-per-million.

PRNU Pixel Response Non-Uniformity.

PSF Point-Spread-Function.

QE Quantum Efficiency.

rms root-mean-square.

RON Readout Noise.

SED Spectral Energy Distribution.

TESS Transiting Exoplanet Survey Satellite.

TP Throughput.

List of Figures

2.1 A representation of a star's flux variation during a planetary full-orbit, taken from Perryman [2014]. 6

2.2 Example of the charge accumulation and charge transfer processes in the CCD. In each sub-figure we can observe the existence of the gate structures, which allow us to create potential wells in each pixel by voltage manipulation. Taken from Howell [2006] 8

2.3 "Bucket of Rain" exemplification of the readout process in the CCD. This image was taken from Howell [2006]. 9

2.4 The CCCD, from the manufacturer's datasheet (e2v technologies [uk]). We can easily distinguish the image frame in the top of the detector and the shielded storage frame in the bottom. 12

2.5 The Cheops' CCD scheme, with a full array representation. Taken from Hoyer [2017]. 13

2.6 Expected Dark Current Rate for different CCD's Temperatures. Obtained from the manufacturer's datasheet. 18

2.7 A CCD's photon transfer curve. In the y-axis we have the Noise and in the x-axis the signal, both in logarithmic scales. We can observe the three different noise regimes over the dynamic range of the CCD - read noise, shot noise and fixed patter noise. 20

3.1 Scheme of the CHEOPS data reduction pipeline, taken from the Hoyer [2017] 24

3.2 Scheme of the CHEOPS input and reference files data, with reference to the different products that are used throughout the reduction process. 26

3.3 Static Bias Sub-Array. Obtained during ground calibration tests of the CCCD. 28

3.4 Individual Image Bias Distribution for a simulated exposure with **3** stacked images. In the top-right corner, it is represented the mean value of the distribution - bias offset, and the standard deviation - RON. 29
3.5 Non-Linearity tests and results. **Left:** Mean signal vs. exposure time curve with the straight line best fit (red) and the saturation level highlighted with the dotted lines. **Right:** Non-Linearity percentage vs the Mean Signal (ADU). 30
3.6 Linearisation curve of the CCCD, with a linear fit for comparison. . 31
3.7 Gain Variation in an observation. In the x-axis we have represented the Modified Julian Date (MJD) of the time each exposure was taken, and its corresponding gain, in e-/ADU, in the y-axis. The plot does not show all the points of that observation for a better analysis on the variation. 34
3.8 Time Map Full Frame, used to estimate the dark current in each pixel. 36
3.9 Dark Rate distribution for a single exposure. In the top-right corner we show the mean dark rate value for this exposure. 36
3.10 Quantum efficiency and Throughput of Cheops, for a range of wavelengths. **Top:** Quantum efficiency. **Bottom:** Throughput 38
3.11 Flux Ratio of both the SED and Black Body curve for a star with T=5170K 39
3.12 Flat Weights for a star with T=5170K, using different methods. In blue we have the weights using the uniform mode. In red the weights for a star with the temperature of the Sun (T=5777K) and using the SED. In green we present the weights calculated using the black body curve and in orange the SED. For this calculation we assume the real temperature of the star to be 5170K. 40
3.13 Final Flat-Field of the full array, computed using the weighted flats mode. The weights were calculated using the SED, for a star with T=5170K. The mean and standard deviation of the image are shown as well. 41
3.14 An imagette (30*30 pix) of an exposure. We can see that it contains the pixels with higher variations of the sub-array. 45
3.15 The output of the aperture photometry module. **Left:** The circular aperture created in the pipeline, centered in the target's computed center of gravity, for a single exposure. **Right:** The flux variation vs observation time, i.e the final light curve. 49

3.16 Bias Distribution of an exposure, with 3 stacked images, for both the normal and reduced cases. **Left:** Normal case without bits reduction. **Right:** Sub-array reduced by 3 bits. 51

4.1 Diagram of the imagette linearisation module. Since each imagette may offset from each other by a small number of pixels, we will only linearise the overlapping pixels, represented in green. Image is not to scale, and it serves as representation only. 55

4.2 Linearisation Correction Curve for low read ADU values. We can observe the existing zero-point offset 56

4.3 Simulated Stacked Truth Data Frame. The Star is simulated as a Gaussian with a variable centroid, amplitude and width, inputed by the user. 56

4.4 Simulated Stacked Bias Frame, using a random normal distribution. The simulation was made with **15** stacked-images. On the top right corner we have the individual image's user inputed bias and RON. . 57

4.5 Linearity Correction Curve and Interpolated Curve, used for adding non-linearity 57

4.6 Point-to-Point difference between a simulated,error-free sub-array and a imagette linearisation corrected image obtained by stacking **15** individual images. The results represent only the aperture pixels, and the values are uniformised - average error per unstacked image. The values in orange represent the pixels inside the Imagette area. The green horizontal line represents the RON and the red line (μ) represents the average point-to-point difference. **a**) Plot with a centroid variation σ of 1.5 pixels. **b**) No centroid variation. **c**) No centroid variation and no RON. **d**) No RON and centroid variation σ of 1.5 pixels. 60

4.7 Point-to-point Flux difference between the Truth Subarray and the Imagette Linearised Subarray, for the aperture pixels. This simulation was made with no RON, with 15 stacked images and centroid variation σ of 1.5 pixels. The simulated star had a σ width of 5 pixels, in each axis. 61

4.8 Linearisation Test Module final results: In blue we have the control line, in orange the imagette linearisation flux ratio and in green the uniform linearisation flux ratio. In the top-right corner we show the mean flux ratio for both situations. **Left:** Simulation made with all three errors present and a centroid σ of 2.5 pixels. **Right:** No centroid variation. Both simulations were made with **15** stacked images. . . . 63

4.9 Linearisation Test Module final results- Simulation with a variable number of stacked images. On the y axis we represent the P2P in ppm. The RON was of 5 ADU rms and we used 25 exposures, with a centroid variation of 2.5 pixels and a star's maximum Amplitude of 64000 ADUs. In orange we present the results given by the original linearisation correction method and in blue the linearisation with imagettes method. 64

4.10 The three different correction curves: **Blue:** Spline Correction Curve. **Orange:** Low ADU-Adjusted Curve. **Green:** Offset Corrected Curve. In the bottom-right of the figure we can observe an augmented region of the correction curves for lower ADU values. 65

4.11 Correction Curve Simulation Results - in the y-axis of each sub-figure we show the P2P in ppm. The nominal values for the number of exposures,number of stacked images, centroid sigma and star's amplitude was 15 images, 15 stacked images, 2.5 pixels and 64000 ADUs, respectively. The blue curve is a simulation with the offseted curve and the orange with the low-adjusted curve. **a):** Varying number of exposures. **b):** Varying number of stacked images. **c):** Changing Centroid's sigma. **d):** Varying Star Maximum Amplitude. 68

4.12 Fractional Imagette Pixels Results. In the y-axis it is represented the Flux Ratio between the measured linearised flux and the truth flux and in the x-axis the Input Fraction. **Top:** We varied the centroid, keeping the other variables static. **Middle:** We changed the simulated star's maximum amplitude. **Bottom:** We varied the number of stacked images. 73

4.13 Fractional Imagette Relative Photometric Results. In the y-axis it is represented the P2P in ppm. The simulation was made with a variable number of stacked images and with nominal values of 25 exposures, centroid's sigma of 2.5 pixels, RON of 5 ADU rms and star's maximum amplitude of 64000 ADUs. In blue we represent the imagette method, without the use of fractional pixels (*Input Fraction = 1*). In green we present the results for an imagette method that uses all the available information (*Input Fraction = 0*). 74

A.1 Pipeline Final Plot Example. We show the light curve, contamination, the RAW and corrected sub-array, the target's center variation, the background and dark rate and the gain. 87

Contents

List of Figures xiii

1 Introduction **1**

1.1 Motivation 1

1.2 Objectives 2

2 State of the Art **3**

2.1 CHEOPS Mission 3

2.2 Exoplanet Detection Methods 4

2.3 CCD 6

2.3.1 CHEOPS CCD 10

2.3.2 CCCD parameters 11

2.3.3 Noise Sources 13

2.3.3.1 Bias and Readout Noise 13

2.3.3.2 CCD and ADC Non-Linearity 15

2.3.3.3 Photon Noise 17

2.3.3.4 Dark Current 17

2.3.3.5 Background Photons 18

2.3.3.6 Pixel-Response Non Uniformities 19

2.3.3.7 Signal-To-Noise 19

3 Original Pipeline **23**

3.1 Pipeline Data Overview 24

3.2 Data Reduction Calibration 25

3.2.1 Bias and Read-out noise Correction 26

3.2.2 Non-Linearity Correction 29

3.2.3 ADU's - Photon Conversion 32

3.2.4 Dark Current Correction 34

3.2.5 Flat-Field Correction 37

3.2.6 Event Flagging . . . 41
3.3 Data Reduction Correction . . . 43
3.3.1 Smearing Correction . . . 43
3.3.2 Bad Pixel Correction . . . 44
3.3.3 Jitter Estimate . . . 45
3.3.4 Pixel to Physical Coordinate Conversion . . . 46
3.3.5 Background Estimate and Correction . . . 47
3.4 Photometric Extraction . . . 47
3.4.1 Simulation and Contamination Estimate . . . 48
3.4.2 Aperture Photometry . . . 48
3.4.3 Light Curve Quality Analysis . . . 49
3.5 Bits Reduction . . . 50

4 Pipeline Modifications 53
4.1 Improving the Linearisation module - Linearisation with Imagettes . . 54
4.2 Linearisation Functional Test . . . 56
4.2.1 Test Validation Criteria . . . 58
4.3 Offset Curve vs Low ADU Correction Curve . . . 64
4.4 Fractional Imagette Pixels . . . 68

5 Conclusions 75

6 Bibliography 77

Appendices 83
A Pipeline Manual . . . 85

1

Introduction

1.1 Motivation

The research surrounding the search and study of extra-solar planets is quickly growing and is defined as one of the main priorities of the European Space Agency (ESA) for the Cosmic Vision 2015-2025 program, with missions such as CHEOPS to be launched in the beginning of 2019, PLATO (PLAnetary Transit and Oscillations of stars) around 2026 and ARIEL (Atmospheric Remote-sensing Exoplanet Large-survey) in 2028. These missions will be fundamental for the comprehension and study of the planetary science field. The role of Charged Coupled Devices (CCDs) in this field is notorious as these detectors have been and will be extensively used in both space and ground-based surveys. Their high efficiency in the optical region of the electromagnetic spectrum is one of the most important factors for the use of this detector. In subsection 2.3 we present the CCD and its main characteristics.

Throughout the years it has been necessary not only to collect and store the information gathered by these detectors, but also to pre-process the collected raw data, removing errors that naturally arise from the detector's and electronics's intrinsic characteristics, as well as from environmental effects. This correction process will later facilitate the consequent job of data analysis done by the science teams. The CHEOPS mission is no exception to this, since a data reduction pipeline, hereafter DRP, has been developed. This pipeline is a software developed in the Python programming language, that will correct the raw images for undesired instrumental and astrophysical signals and then extract the light curve of the target star. The light curve represents the variation of the flux received from a star with time. It may be used in exoplanet science to study the flux changes created by a planet transiting its host star.

1.2 Objectives

The goal of this dissertation is to study and test the CHEOPS DRP developed by the Institute of Astrophysics (IA), more specifically by the Center of Astrophysics of the University of Porto (CAUP) and by the Laboratoire d'Astrophysique de Marseille (LAM), both institutional members of the CHEOPS science team. Each of the pipeline's components are studied and described, with an emphasis being given to the calibration module of this data reduction software.

In chapter 4, I present the development of a parallel pipeline and the creation and testing of alternative algorithms for it. The reasoning behind creating an alternative pipeline is to simplify the testing of these new algorithms, whose goal is to improve the performance of existing modules. The original pipeline was developed with software design restrictions from the CHEOPS consortium and ESA. These restrictions make the code more robust and flawless, but hinder the module's development.

The final result of this thesis work will be therefore, a fully functional replicated pipeline, with an addition of alternative methods compared to the original one.

2

State of the Art

2.1 CHEOPS Mission

The CHEOPS mission is a small class mission and the first low-budget mission funded by both ESA and by Switzerland. It is therefore a mission with constraints on many factors such as the payload, its weight and mission implementation time (<4 years, when most astrophysics space missions are developed over 10 years or more) Redbook [2013].

The CHEOPS satellite is a lightweight satellite with less than 100 kg and with a small field-of-view, capable of ultra-precision photometric measurements. The satellite will have a mission duration of approximately 3.5 years. From these, 20% of its observation time will be on request and open to the scientific community.

The CHEOPS mission payload is constituted by a telescope with 33 cm of aperture, with a CCD detector in the focal plane. The definition of a CCD and its characteristics can be found on subsection 2.3. The CHEOPS satellite will follow a Sun Synchronous Orbit with an altitude in the range of 620-800 km.

The mission's main goal is to get tighter constraints on the characteristics of already know exoplanets of which we may (or not) already possess their mass. CHEOPS will contribute to provide tight constraints on exoplanets' density allowing to infer the existence (or not) of an atmosphere. CHEOPS will help to unveil the formation and evolution of exoplanets from the rocky Super Earth (1-10MEarth) to the gaseous Neptunes (10-30 Mearth) and Jupiters (30-1000MEarth). It will, additionally, study the creation process of these Neptunes, and provide candidates for future studies with subsequent missions in terms of atmospheric characteristics .

Most of the planetary systems which will be subject to study are targets which already have their mass well constrained from Doppler-surveys (radial velocity method of detection, which is outside of the scope of this work, for a description of this method see Perryman [2014]. The targets of this mission will be stars brighter than the 12th magnitude on the visible (V) band. This band has a mean wavelength of 450 nanometers (nm). The CHEOPS satellite is expected to be able to make observations with precisions in the order of 100 parts-per-million (ppm), in 1 hour of integration time, depending on the star's brightness. 100 ppm will be the expected precision for the case of a star in the 12th Magnitude. If we compare this precision with the precision obtained in transit ground surveys, like the Wide Angle Search for Planets (WASP) or the Next Generation Transit Survey (NGTS), which have a precision of more than 10000 and 1000 ppm in 1 hour integration time, (Christian et al. [2006] ,Wheatley et al. [2018]), respectively, we can see the performance potential of space-based surveys. This precision boost occurs since space-based surveys' measurements are not affected by the Earth's atmosphere, which is a crucial factor for high-precision photometry.

For a small read on the CHEOPS satellite's main characteristics I point the reader towards Fortier et al. [2014]. In this paper there is an overview of the mission as well as a small description of its main components and objectives. For an extended read, the already quoted Redbook [2013]is recommended.

In Gaidos et al. [2017] the expected performance of the CHEOPS satellite is compared with the NASA'S Transiting Exoplanet Survey Satellite (TESS). It is evaluated each mission's advantages in terms of observational performance relative to the other and how they can complement each other in their findings. Since each of these missions' detectors are more responsive to certain wavelengths (blue for CHEOPS and red for TESS) there may be some spectroscopic information coming from inter-comparing both of these missions results. CHEOPS will thus be a crucial follow-up mission to TESS.

The expected launch date of the CHEOPS mission is in the beginning of 2019, but may be subject to change.

2.2 Exoplanet Detection Methods

There are many different methods used in exoplanet detection. The first detected exoplanet around a solar type star was discovered using the radial velocity technique Mayor and Queloz [1995], explained as follows: the planet may perturb the star's movement around their common center of gravity of the system, and we may observe a displacement in the spectral lines of the light received from the star due to the

Doppler effect. From this displacement, we can measure the radial velocity of the star, and its change may provide us with the information of the minimum mass of the perturbing planet, due to the lack of knowledge regarding the orbit's inclination. This technique is used in many ground surveys.

$$\Delta F = (\frac{R_p}{R_\star})^2 \tag{2.1}$$

$$b = \frac{a * cos(i)}{R_\star} \tag{2.2}$$

Another method generally used in planet detection surveys, and the one employed in the CHEOPS mission, is the transit method, explained hereafter:

This method consists in the observation of a planet's host star, looking for the passage of a planet. Whenever the planet enters the observer's line of sight, the amount of light received from the star diminishes by a small amount, for a certain period of time. This flux variation is dependent on the radius of both the host star and the planet (equation 2.1). The duration of the planetary transit highly depends on the planet's size and in other factors such as the planetary period of the orbit and the impact parameter, which is the projected distance between the planet and the star's center during mid transit Perryman [2014]. The impact parameter can be computed using equation 2.2, where a is the semi-major distance and i the orbit inclination. The equations 2.1 and 2.2 where taken from Perryman [2014].

In conclusion, by monitoring the variations of the stellar flux versus time, we can access the ratio of the planet radius over the stellar radius. We can also obtain the planet's period by observing successive transits. The first planetary transit was observed for the star HD 209458 by Charbonneau et al. [2000].

$$\rho_{mean} = \frac{M_p}{{R_p}^3} \tag{2.3}$$

Getting information from the radius and mass and consequently the mean density (equation 2.3) gives us precious information in order to characterize the planet as well as making assumptions on its core composition.

In addition to the transit we can sometimes observe an occultation phenomenon which is characterized by a small diminution of the flux when the planet is behind the star, therefore not visible to us. This happens because the planet stops reflecting its host star's light towards us. The flux variation is smaller than in a transit but can still be noticeable and quantifiable.

Furthermore, by analysing transit time variations, the existence of other planets in a planetary system can be inferred e.g Gillon et al. [2017]. This method predicts that the existence of many planets in a system causes mutual gravitational interaction on each of the planets, causing small variations on the observing time of the transit.

In figure 2.1, we can observe a representation of the flux variation from a star, during a full planetary orbit. We can see that the light received by the star diminishes by ΔF during transit and slightly lowers during the secondary eclipse (occultation). Just before/after occultation, the flux is at its peak, due to the contribution of light reflected from the planet's atmosphere towards the observer. The transit's impact parameter is also represented in this figure.

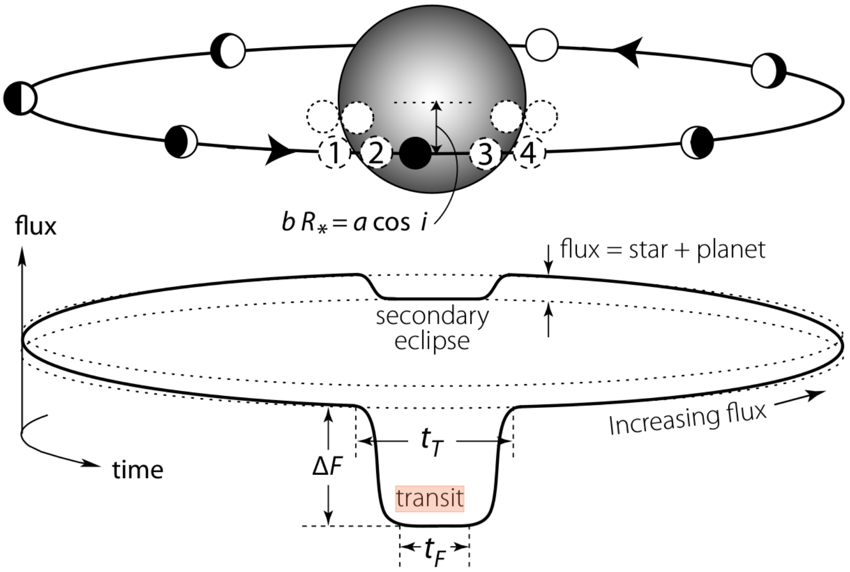


Figure 2.1: A representation of a star's flux variation during a planetary full-orbit, taken from Perryman [2014].

2.3 CCD

The detector used in the CHEOPS satellite is a CCD.

The CCD is an imaging device that works by collecting electric charges, created by the incidence of photons on its photo-active region (pixels). It is therefore a pixelised array detector. These type of photon detectors have been used extensively in the last decades and have suffered an exponential growth performance-wise in recent years. Due to their characteristics and reliability they are widely used in Astronomical observations.

The CCDs pixels are usually made from silicon, a semi-conductor, whose energy band gap, between the conduction and valence band, is in the order of 1.14 electron-volts (eV), which means that it absorbs photons in the 1-4eV range Howell [2006]. The conduction band is an unfilled energy band in an isolator or semiconductor material, where electrons, if belonging to it, may move freely and create an electric current if an electric field is present Tavernier [2000]. On the other hand, the valence band is typically the highest band of energies where electrons are present at the absolute zero temperature. In metals, these two bands overlap and in insulators this gap is very high. In semi-conductors' the small energy band gap makes them great photon detectors. Since this gap has the adequate size in semi-conductors (not too big/small), we can control the number of charges in the conduction band.

Each of the CCD's pixels is mainly composed of a silicon substrate, a dielectric interface of typically silicon dioxide and a metal structure called a gate.

Operation:

As previously referred, CCDs work by accumulating electrons in its pixels. During an observation, the amount of charges in the detector will be proportional to the number of photons absorbed on its layers. The photon absorption and charge accumulation processes occur in this fashion:

By biasing the detector, we deplete it of mobile charges, creating a so-called depletion region. By increasing the applied voltage we expand the depletion region and an ionizing photon can create free charge in it. When a photon is absorbed by the semi-conductor material, a free-electron charge and a positive hole are created. The electron will move from the valence to the conduction band. If a voltage was applied to the gate structure, we will manage to push the electrons and holes to opposite sides of the silicon sub-structure. The free electron charge will thus be collected due to the voltage applied, since the electrons will go towards the highest potential.

Because each pixel typically has three gate structures that allow us to manipulate 3 different potentials, we can use these gates to create potential wells in each pixel, which will permit us to keep the electrons from freely moving to other regions of the detector. This will let us control the charge transfer, from pixel-to-pixel, during the readout process.

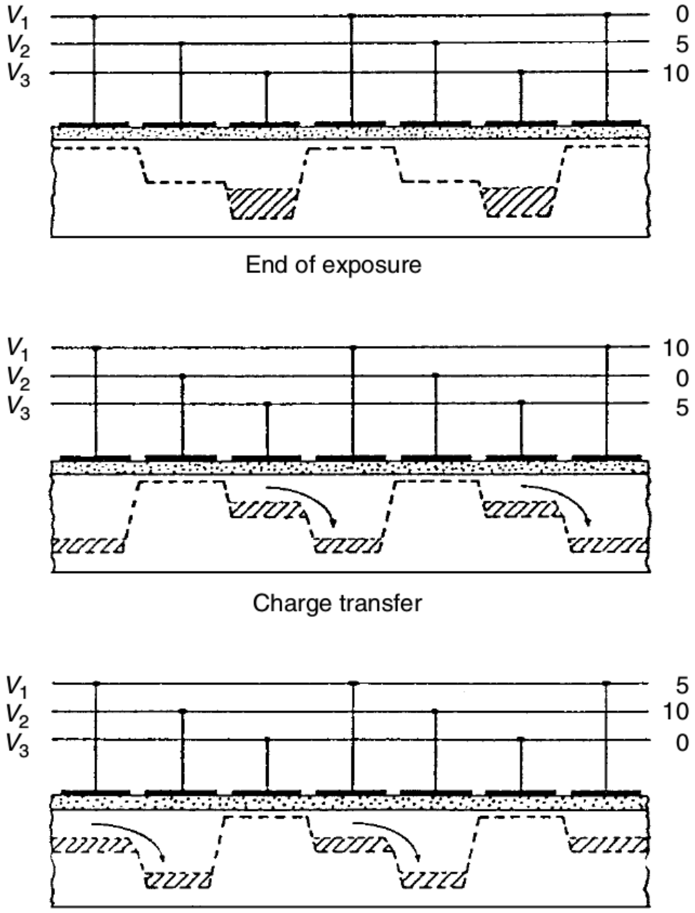


Figure 2.2: Example of the charge accumulation and charge transfer processes in the CCD. In each sub-figure we can observe the existence of the gate structures, which allow us to create potential wells in each pixel by voltage manipulation. Taken from Howell [2006]

.

The general charge accumulation and readout process in the CCD happens as follows:

- At first, we accumulate charge in each pixel for a certain period of time (exposure time) - incident photons in the photo-active region of the detectors create free-electrons, which are kept in place by manipulating the gate's voltages.

- When the exposure ends, and the readout process begins, each pixel's charge will be sequentially shifted to the neighbouring pixel by manipulating the gates' voltages, in each clock cycle. An example of the charge pixel-to-pixel transfer is present in figure 2.2. In the top sub-figure, a potential well is created in each pixel (we can see two potential wells), with the use of voltage manipulation. At this point the CCD is accumulating electron charge in its pixels. In the middle sub-figure, the readout process begins and the charges collected in the first pixel, with the use of the voltage shifting technique, start to be transferred to the neighbouring pixel. In the bottom sub-figure, we can see how the process continues by shifting the potentials once again.

- The charge packet will be transferred row by row, until it reaches the register, that is, in some cases, at the bottom of the CCD. An example commonly used to explain the readout process of the CCD is shown in figure 2.3. In this figure, the buckets represent the potential wells and the drops represent electrons. The electrons get transfered row by row until they reach the CCD's register (vertical board in the figure) and each "bucket" will be measured one-by-one.

- When the charge packet reaches the output gate, the amount of charge is sampled and converted to a voltage. Afterwards, this voltage will be converted to a digital number: analogic-digital units, (ADU) using an electronic circuit - Analog-to-Digital Converter (ADC).

The amount of electrons needed to produce one ADU is called the gain of the circuit. A gain of 10 electrons/ADU means that you need 10 electrons to have a digital output number of 1. In the CHEOPS pipeline we have the gains coined in terms of photon-electrons/ADU. For more information on gain and its role in the pipeline, refer to subsection 3.2.3.

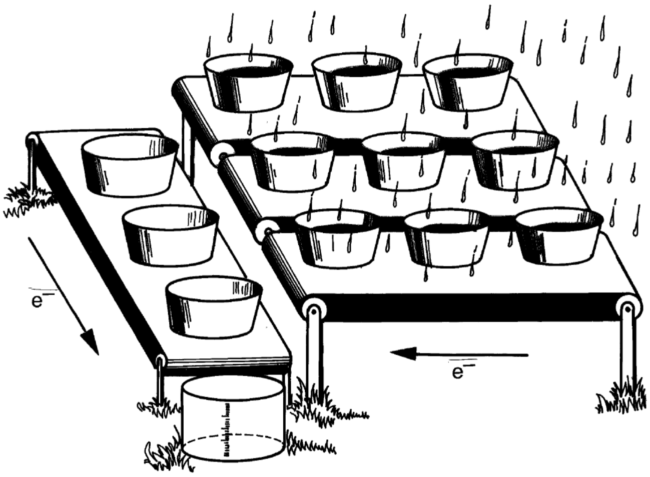


Figure 2.3: "Bucket of Rain" exemplification of the readout process in the CCD. This image was taken from Howell [2006].

2.3.1 CHEOPS CCD

The detector of the CHEOPS mission is a frame-transfer, thinned back-side illuminated CCD.

This type of detector has one frame exposed to the incoming radiation and another which is shielded from external light. In figure 2.4 we can see a picture of the CCD used in the CHEOPS mission. The two different frames can be observed.

The detector is a back-side illuminated CCD which means that incoming photons hit the detector from the back surface of the detector (instead of the front in the front-sided versions) and thinned which means that the silicon material's thickness was reduced to a small amount (<15 microns), during the manufacture process.

Some telescopes have a controllable shutter that can be opened/closed as required. The existence of a shutter allows the observation and readout processes to be done with the same set of pixels, since we can block the incidence of external photons. The CHEOPS telescope has no shutter, so we should not use the same group of pixels for the observation and readout processes. Therefore, after each individual exposure, the object frame is quickly transfered to the non-exposed bottom pixels and read by the CCD's electronics. This transfer comes with errors associated (e.g smearing, explained in subsection 3.3.1) but it comes with advantages, more specifically, it has the ability of almost continuous observation, as we just need a small amount of time for frame transferring. Non-frame transfer CCDs imply that observation and readout can't be done simultaneously, which implies a small gap between two exposures .

After the transfer, the data is subsequently read by an amplifier at the bottom of the storage section. Each row is then read one by one as the rows are shifted down after the previous row has been fully read, as was exemplified back in figure 2.3 with the bucket of rain example.

This detector has a higher active region compared to the front-sided versions. The reason is that in the latter, the gate structure blocks part of the pixel's sensitive area, greatly reducing the Quantum Efficiency (QE) of the detector, further explained in subsection 3.2.5. This CCD has thus a better performance and higher efficiency. As drawbacks it has a higher manufacturing cost and possible nonuniform thinning leading to surface and pixel response non-uniformities (PRNUs), as well as smaller well depths.

In figure 2.5, we can observe a schematic of the CCCD, where there are represented both its image and storage section. They are also identified the dark, overscan and blank rows and columns. In the bottom of the storage section, the left and right amplifier are also shown, although only the right one is used for readout, since the other will be used as a backup if the first fails during the mission's duration.

The active region of the image section of the CCCD is 1024*1024 pixels and the active storage section is 1024 (H) pixels for 1033 (V). The extra nine rows are for both the top 6 overscan rows, used for smearing correction, and for the 3 dark top rows, which may be used for the correction of dark current, explained in subsection 2.2.3.

To be more precise, around both the image and the storage sections there are 16 columns on each side, named dark left and right columns, which are used for dark current correction. These are shielded physical pixels that don't collect external radiation. There are, as well, 4 virtual columns of pixels for overscan purposes on the left and on the right of the frame, which are used to calculate the bias and read-out-noise (subsection 2.3.3). The detector is operated at -40°C in order to minimise the dark current present. In the bottom of the storage section we can observe the register (white row).

Finally, in the image section, we can observe a 200*200 pixels sub-array frame, containing a target star.

2.3.2 CCCD parameters

We present some parameters that may be important in order to characterise the CCCD and understand how it may behave in terms of performance.

A relevant characteristic in CCDs is the Charge Transfer Efficiency (CTE). It represents the percentage of electrons that don't get transferred in each pixel transfer, during the readout process of the detector. A CTE of 99% means that on average, for each transfer, one out of 100 will not be transfered. A good CTE is important as not to leave behind a great number of electrons during the transfer and readout operation, decreasing the total error. Nowadays CCD's have very high CTEs, so it is not a considerable source of error. In the case of the CCCD the CTE is around 99.999% according to the CCD's manufacturer's datasheet (e2v technologies [uk]), making this factor almost negligible in the CCCD's performance.

The full well capacity of a CCD is a crucial parameter to be aware of. It represents the maximum number of electrons that a pixel's potential well can hold. It is

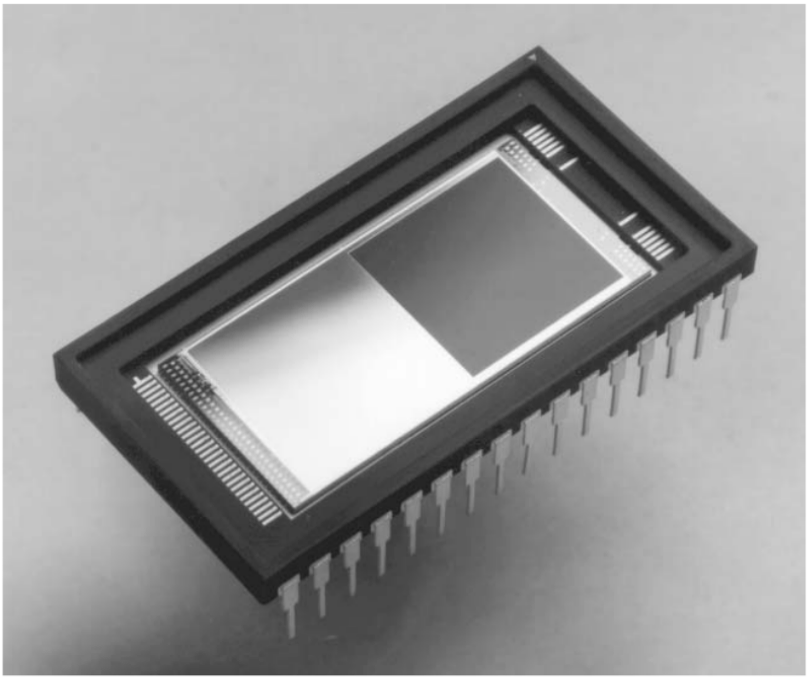


Figure 2.4: The CCCD, from the manufacturer's datasheet (e2v technologies [uk]). We can easily distinguish the image frame in the top of the detector and the shielded storage frame in the bottom.

dependent on factors such as the pixel's size. The CCCD has a full well capacity of $1.32 * 10^5\ e^-$.

The plate scale is another important characteristic that connects the angular separation of an object with the linear separation of its image at the focal plane. The plate scale value should vary depending of the application. We can generally approximate it in the following way (taken from Howell [2006]):

$$\text{Plate Scale} = \frac{206265 * \text{pixel size}}{1000 * f} arcsec/pixel \tag{2.4}$$

where f is the focal length in mm. In case of the CHEOP's satelitte we have an effective focal length of 2681 mm and a pixel size of 13 microns, therefore we obtain a plate scale of approximately 1 arcsec/pixel. According to Fortier et al. [2014], the plate scale of the CCCD is equal to 1 arcsec/pixel, therefore this equation is, in this situation, adequate to estimate the plate scale.

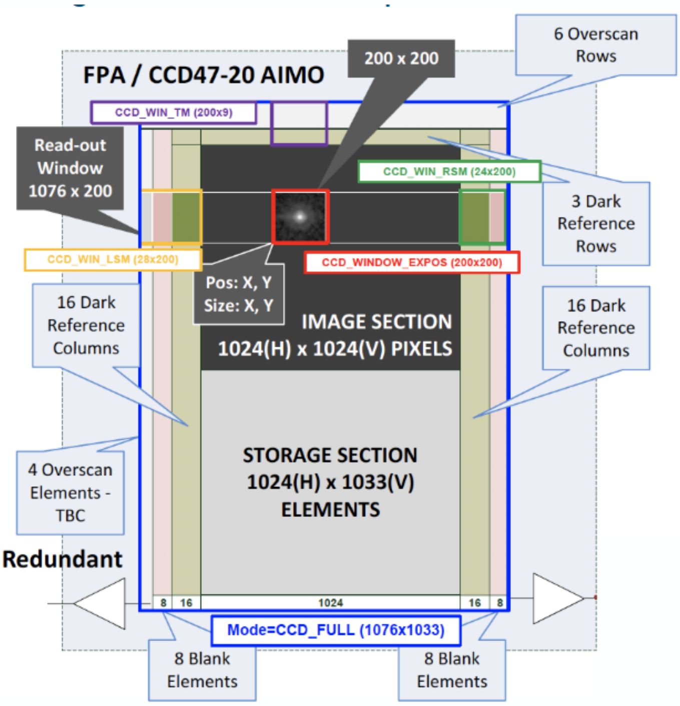


Figure 2.5: The Cheops' CCD scheme, with a full array representation. Taken from Hoyer [2017].

2.3.3 Noise Sources

In the current subsection we analyse some of the most relevant noise sources that affect the CCD's and its electronics. It is crucial to characterise the overall noise of a CCD system in order to understand how it will behave in observational situation.

2.3.3.1 Bias and Readout Noise

The Bias or zero-level is an offset created by the CCD readout circuit. During the readout process, a bias voltage is added to the measured charge voltage, before the conversion from an analog (volts) to a digital signal (ADU). This offset value is added to the charge signal to avoid having negative ADU read values after the CCD readout process, as a negative signal will use an extra bit of information (sign-bit), so one additional bit value will have to be used for representation, which is unwanted, as valuable information may be lost or .

Bias is thus an offset voltage value created by the CCD's electronics. When there is no photon incidence in a pixel, the readout value for that pixel will correspond to this bias level in ADUs, added by the electronic circuit, deviated by a small number of ADUs. This deviation is called the read-out-noise:

The read-out-noise (RON) is the amount of noise, in electrons root mean square (rms) per pixel, that is added to the read signal by deviations in the analogic to digital signal conversion and from added electrons by the electronic components. The RON depends then on the characteristics of the CCD and the reading electronics. It sets a noise floor to the device.

There are two different ways we can compute the bias and, as a consequence, the RON:

- By the use of bias frames - 2D images obtained for an integration time of 0 seconds, without exposure to external radiation, i.e with the telescope's shutter closed.
- By using overscan columns - over-clocked columns of pixels, which are virtual pixels present on the CCD, generated by sending additional clock cycles to the CCD output electronics.

In both cases, we can measure the RON by creating a distribution of the bias values of either the bias frames or the overscan pixels' values and calculating the standard deviation of this distribution. The RON, measured in electrons/pixel, is equal to the standard deviation of the distribution times the gain of the circuit. As previously explained, the gain is the amount of electrons needed to produce an output digital signal of 1.

$$\text{Read Noise} = \sigma_{ADU} * \text{Gain} \tag{2.5}$$

The bias and RON follow, most of the time, a Gaussian distribution where the mean value is equal to the bias and the standard deviation to the RON.

It is important to note that throughout the years, with technological advances, CCDs have been significantly improved, drastically reducing the amount of noise intrinsic to them. In recent CCDs, the RON can contribute just a small error of less than 10 e- rms/pixel.

According to the CCCD datasheet we will have, for a readout rate of 20kHz, an expected readout noise of 2e-rms/pixel, but with an higher readout rate this value will increase (e2v technologies [uk]). This occurs since an higher readout speed will cause higher thermal swings, which in turn affect the uncertainty in the measured charge signal. This means that faster readout rates will come with an expense on the amount of noise but as an advantage it permits us to readout the CCD's pixels faster.

By having the information on the RON of the detector we can thus quantify the dynamic range of the CCD. It can be defined as the range of values on which the CCD operates better in. It can be approximated by the following formula:

$$\text{Dynamic Range} = \frac{\text{Full well capacity}}{\text{Read noise}} \tag{2.6}$$

In this equation the full well capacity is equal to 132 000 e-, in the case of the CCCD. The read noise is, according to the manufacturer's datasheet, equal to 2e-/pixel, for a CCD's readout rate of 20kHz. Therefore, using these given values we can assume that the CCCD has a dynamic range of 66000.

2.3.3.2 CCD and ADC Non-Linearity

There should be a proportionality between the number of electron charges generated in a CCD and the number of photons that strike the detector Janesick [2001]. A linear CCD would display a linear relation between the output signal and the number of incident photons. One of the main advantages of CCDs is their linearity. At a first order, CCDs are linear detectors. Of course, there are small deviations that for high precision analysis need to be corrected. These deviations to linearity need to be studied and quantified. There are many techniques which may be used to perform this characterisation, for example: a method where the CCD observes a star with a known brightness. A plot is made of the obtained read ADU values in function of the exposure time. We expect, in the linear case, by doubling the exposure time, we will also double the amount of photons collected.

There are two types of non-linearity present in a CCD detector:

- Differential Non-Linearity
- Integral Non-Linearity

The differential non-linearity is an important factor to take into account. Since we use a curve approximation to represent discreet levels on the conversion of an analog (voltage) to a digital signal, we expect a small error to arise, since fractional values will be approximated by excess or defect. We therefore have a maximum uncertainty of $\pm 0.5 ADUs$ for each level.

This type of non-linearity is also called the digitization noise, which is equal to the gain of the CCD times the one sigma error for using discreet levels, which is around 0.289, according to Janesick [2001].

The other source of non-linearity is the integral non-linearity which is measured, for a certain ADU level, as the departure of the detector's response from the linear relationship, according to the ground calibration team. This non-linearity is studied and corrected in the pipeline, in subsection 3.2.2.

Signal Representation:

As explained previously, during the charge conversion process we convert an analog to a digital signal, using an ADC. The number of bits of the ADC in the readout circuit gives us the maximum digital number (DN) that can be represented during the readout process:

$$\text{Maximum DN} = 2^N - 1 \tag{2.7}$$

For the CCCD's ADC, where the number of bits is 16, the maximum output DN is 65535. Unfortunately, there are different factors which may further limit the maximum ADU value that can be used for representation, which are important to study and quantify, they are:

- Pixel Saturation
- ADC Saturation
- Non-Linearity

The first type is related to the maximum-well capacity of the CCD's pixels. Whenever the number of electrons in a pixel exceeds the full-well capacity, pixel bleeding will occur and charges leak to the neighbouring pixels, causing errors in the observational results. In order to stop this from happening, anti-blooming gates can be used in the manufacture of the CCD. These gate structures drain the saturated pixel stopping it from leaking charges into neighbouring pixels. In the case of the CCCD there were not used any blooming gates, which means that extra care needs to be taken to stop saturation from happening - shorter exposures for brighter targets. On the upside, the CCCD will have a higher sensitivity as the lack of these gate structures will result in a bigger active pixel region.

ADC saturation means that larger than $65335 * \text{gain}$ electron read values can't be converted to their corresponding digital number, creating visible errors in the data, such as flat-topped stars.

Lastly, we have non-linearity, that can only be detected by observational pre-tests done with the CCD detector. Due to it being impossible to directly detect from the output data, this is the most important of the three mentioned factors to study and quantify. The best course of action to prevent it is to control the exposure time of each Observation - stacking more images with shorter exposure durations for brighter stars, or the reverse for fainter ones, in order to keep the read values in the linear region of the CCD.

2.3.3.3 Photon Noise

Photon Noise is related to the uncertainty on the arrival of photons in a photon detector. It is therefore independent on the CCD and its electronic circuit.

This type of noise follows a Poisson distribution where the one sigma error (σ), hence the noise of an observable, is equal to the square-root of the mean value (only for high values we can make this approximation). If, for example, the CCD reads a signal of 10000 electrons in its pixels, the noise associated with this measure is equal to the square-root of the number of read electrons, therefore the noise in this situation is equal to 100 e-.

2.3.3.4 Dark Current

Dark current is created by thermally generated free electrons (dark electrons) which cause Poisson fluctuations in the read signal. The number of electrons created is equal to the sigma squared of the Poisson's distribution:

$$\sigma^2 = N_{photo-electrons} = \frac{Q_d * t}{e} \tag{2.8}$$

In this equation, Q_d is the dark electrons rate measured in $A/pixel$, t is the exposure time and e the electron charge. As seen by equation 2.8, the dark current changes linearly with the exposure time.

By having a CCD operating at non-zero temperatures ($T \neq 0$ K), thermally generated electrons will introduce an error in the collected astronomical data. In equation 2.9, taken from e2v technologies [uk], we present an expression for the variation of dark current with the CCD's operating temperature:

$$Q_d = Q_{d0} * 1.14 * 10^6 * T^3 * e^{-9080/T} \tag{2.9}$$

Q_{d0} is the dark current rate at T=293K (=20°C) and T the CCD's operating temperature. The dark current rate is, consequently, highly dependent of the CCD's operating temperature.

In figure 2.6 we present this dark current variation for different CCD's temperatures. We can observe that at T = 40°C, we expect a dark rate of more than 1000 electrons/pixel/s, which, in an observational situation, will highly increase the overall noise in the collected data. In contrast, a CCD operating at a temperature of -20°C will be subject to a dark current of only 1 electron/pixel/s.

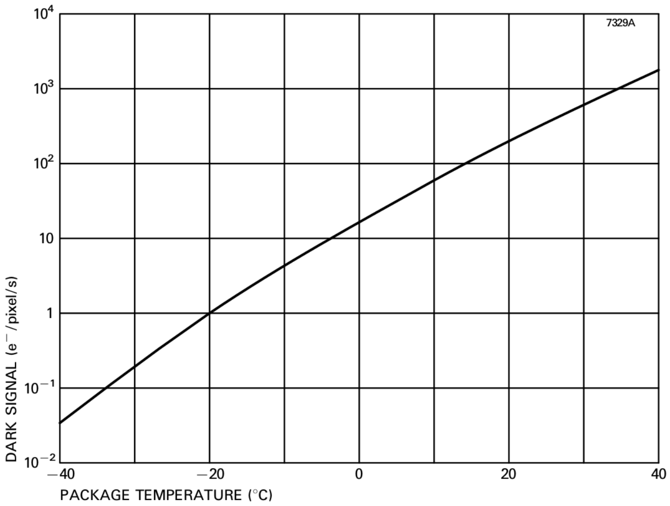


Figure 2.6: Expected Dark Current Rate for different CCD's Temperatures. Obtained from the manufacturer's datasheet.

As can be concluded by analysing both the equation 2.9 and figure 2.6, the CCD should be operated at the lowest possible temperature, in order to minimise the production of dark electrons. Therefore the CCCD will be operated at -40°C which will considerably lower the dark current present. At this temperature, we will expect a dark current of less than 0.1 e-/pixel/s. The method used for correcting dark current in the DRP is explained in subsection 3.2.4.

2.3.3.5 Background Photons

Background Photons come from sources other than the observational target, which may contribute to the total flux measured in the detector. An important source of background photon noise that may affect the results is, for example, stray-light coming from the Earth's or the Moon's surface.

It is important to devise methods that can both estimate this error source and manage to eliminate its contribution, as much as possible, from the results.

We describe, in subsection 3.3.5, the method used in the DRP to estimate and correct this error source.

2.3.3.6 Pixel-Response Non Uniformities

CCD's pixels are subject to small response non-uniformities originating from the manufacture process. Characterising the CCD for these type of pixel non-uniformities is quite important, so to have insight on the response behaviour of each of the pixels when irradiated by photons and consequently the photon flux that is received by each pixel.

Pixel response non-uniformities are wavelength dependent therefore a complex and precise method is necessary to correct them.

The way to study and characterise these pixel-to-pixel variations is called flat-fielding. This process consists in illuminating the full frame CCD with uniform monochromatic light and observe how each of the CCD's pixels will respond. These recorded images are called flat-fields and they contain the information of pixel-to-pixel variations. Typically flat-fielding is done with a single observation, in a certain wavelength, but since in the CHEOPS mission we want to perform high-precision photometry, it is important to use flat-fields with different light wavelengths as each detector can be more/less responsive to a certain region of the electromagnetic spectrum and also the observed targets are not spectrally uniform.

By possessing the information contained in these flat-field images we can then begin to correct the data for these non-uniformities. In subsection 3.2.5 we explain with detail how these non-uniformities were corrected using the flat-fielding procedure.

2.3.3.7 Signal-To-Noise

During the study and characterizations of the CCD it is crucial to understand how the noise components will affect the different observations which will be performed with the detector. It is imperative to quantify each of the noise components that were already mentioned in this work (RON, dark,etc) for a calculation of the ratio between the signal we want to measure and the noise associated with it.

In figure 2.7 we present a generic photon transfer curve of a CCD. It shows a plot of the Noise vs the Signal, both in logarithmic scales, obtained by illuminating the detector with different levels of uniform light. We can observe the three different noise regimes - read noise, shot noise and fixed patter noise. As previously explained, the read noise sets a noise floor for the CCD, therefore it is invariant with the signal. Shot Noise is representing the photon noise of an observable, which rises as the square-root of the signal, hence the 0.5 slope. Finally, for higher signal values, the fixed pattern noise (PRNUs) dominates since it is proportional to the amount of signal. The signal's maximum is reached when we reach full-well capacity.

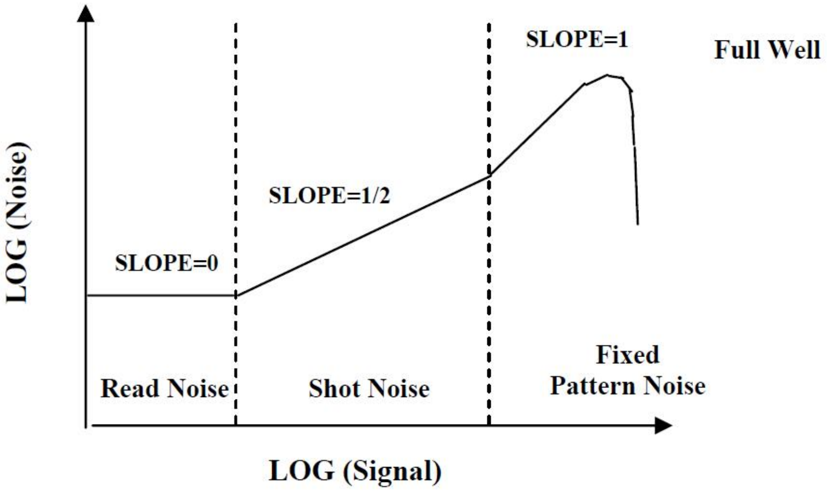


Figure 2.7: A CCD's photon transfer curve. In the y-axis we have the Noise and in the x-axis the signal, both in logarithmic scales. We can observe the three different noise regimes over the dynamic range of the CCD - read noise, shot noise and fixed patter noise.

We can approximate the signal-to-noise ratio of the CCD with the following formula, taken from Howell [2006]:

$$\frac{S}{N} = \frac{N_*}{\sqrt{N_* + n_{pix}(1 + \frac{n_{pix}}{n_B})(N_s + N_D + N_R^2 + G^2\sigma_f^2)}} \tag{2.10}$$

Where N_* is the total number of photons collected from the observed target. n_{pix} is the number of pixels of the aperture used for the calculation of the signal. n_B is the number of pixels used to calculate the background photons, N_S is the number of background photons per pixel, N_D is the number of dark current electrons per pixel, N_R is the total number of electrons per pixel coming from the read noise and $G^2\sigma_f^2$ is the CCD's gain squared times the one sigma noise of a single ADU step (A/D noise), squared - digitization noise.

When all the factors to the right of N_* in the denominator are negligible, the signal-to-noise ratio becomes equal to $\sqrt{N_*}$. This type of situation is called the photon-limited situation. This is a good approximation for observations of bright sources, since the number of received photons, thus the signal, will be quite high compared to the other factors. The same won't happen for fainter sources, where we need to take the other error sources into account for a more precise signal-to-noise estimation.

This equation shows that, in a photon-limited case, for one to increase the signal-to-noise ration by a factor of two, one needs to increase the number of photons by 4. If we consider that the CCD is fully linear, then to double the S/N we need to increase the exposure time by 4.

PRNU and integral non-linearity errors are not present in this equation as they are assumed to be properly corrected in the pipeline.

3

Original Pipeline

In this chapter we present the CHEOPS data reduction pipeline that has been developed by researchers from both CAUP and LAM. The DRP was developed using a modular approach, where each sub-module has a specific function/goal. It was developed using the Python programming language.

The CHEOPS pipeline has three main modules:

- Data Reduction Calibration
- Data Reduction Correction
- Photometric Extraction

The first two are composed of various sub-modules whose objective is to perform a reduction of the raw data's error as well as inherent CCD's error sources. The first one specifically corrects for CCD/electronics systematics such as bias/RON, dark current etc. The second one corrects for environmental undesired effects, such as background, jitter, etc. The last one produces the light curve, calculates its quality, estimates additional sources of error, as well as stores the processed information in FITS files, which is a file format commonly used to store Astronomical data. In the following subsections we do an overview of each module's main functions and explain the logic behind.

A scheme of this pipeline can be found in the figure 3.1. In this figure we can see that the raw data goes through each of these modules, being corrected in the process. In this figure we can also note the mention to the Reference Files, which may be variable test data, that may change through time, during planned calibration measurements throughout the mission's duration. This information is stored in FITS files and will be used throughout the pipeline for correction and calibration purposes, explained more in detail later in the work. The housekeeping information contains auxiliary information regarding the temperature of the many components of the satellite (CCD, electronic chain, etc) as well as the different voltage values of the output circuit.

The two outputs of the pipeline can be seen in the figure: a report of the data reduction process, which contains information collected from the whole reduction, and the final light curve.

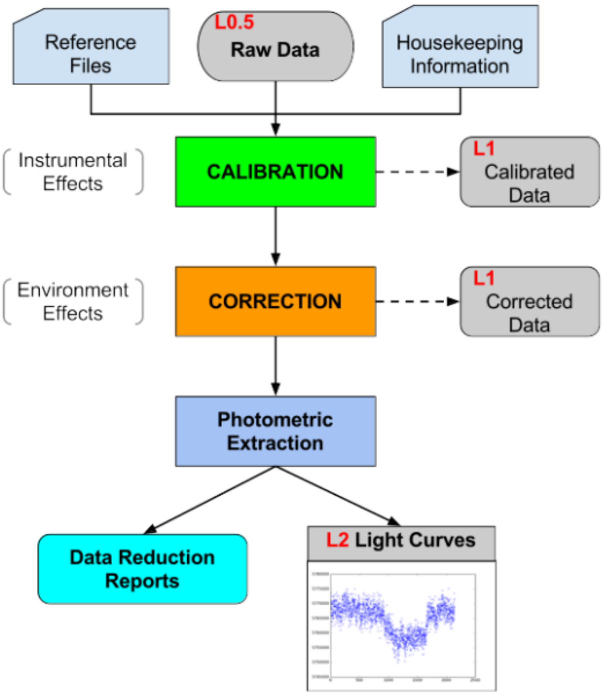


Figure 3.1: Scheme of the CHEOPS data reduction pipeline, taken from the Hoyer [2017]

There is an extra module in the pipeline, called dr_tools which is a library of functions which are used throughout the pipeline. It is separated from the other modules to keep the code clean and organised.

The simulated data, used for the development and testing of the CHEOPS' data reduction pipeline, was created using a software developed by researchers in Switzerland, called CHEOPSim. This program permitted us to create different types of simulated data, according to our needs.

3.1 Pipeline Data Overview

In figure 3.2, we present a schematic of the different products that are used throughout the pipeline. Each data-set folder is composed of a RAW sub-array product, stored inside a FITS file, belonging to an observation. This folder contains the information of the sub-array's RAW data for that observation, the overscan pixel information (overscan rows and columns, used for Bias/RON correction and smearing, subsection 3.3.1), the dark pixels (used to estimate the dark current , subsection 3.2.4) and the metadata, which contains voltages, temperatures and general information of that observation. The RAW data is a set of $N_{\text{exposures}}$ which are each composed of $N_{\text{imagettes}}$ Individual/Unstacked Images. The number of stacked images will depend on the brightness of the observed target, and is between 1 and 60 images.

The data-set folder also contains the Imagettes, where the number of imagettes for that data-set $N_{\text{imagettes}} = N_{\text{stacked}} * N_{\text{exposures}}$. We also possess auxiliary information, from which we can note a file with the already known bad pixels positions (mentioned in subsection 3.3.2), attitude and centroid products - contain information on the spacecraft's position, declination and pointing position (used for pixel to physical coordinates conversion, subsection 3.3.4) and the star catalogue (used for the contamination sub-module, subsection 3.4.1).

In addition, we present the individual calibration products from the Reference Files folder, which contain the static bias (explained in subsection 3.2.1), the Splines used for non-linearity correction (subsection 3.2.2), the Gain Product to estimate the gain of the CCD (subsection 3.2.3), the Quantum Efficiency (QE), Throughput (TP), Spectral Energy Distribution (SED) and Flat-Fields (FFs) for the PRNUs correction (subsection 3.2.5), the Line-of-sight (LOS) and Temperature constraints used in the event flagging sub-module (subsection 3.2.6) and finally the Point-Spread-Function (PSF) used in the contamination estimate module (subsection 3.4.1).

Finally, we want to note that this work focuses, in its majority, in the first part of the pipeline, the calibration. In view of this, the other two sub-modules are explained with less detail. Also, the pipeline is still in development and we may expect further updates in the future. Therefore, the version presented in this work may not correspond to the final version as it was up-to-date only at the time of the writing of this work.

3.2 Data Reduction Calibration

This module's objective is to collect the RAW data already stored in FITS files and to calibrate it, taking into consideration errors associated with the instrumentation. Included in this module are the following sub-modules:

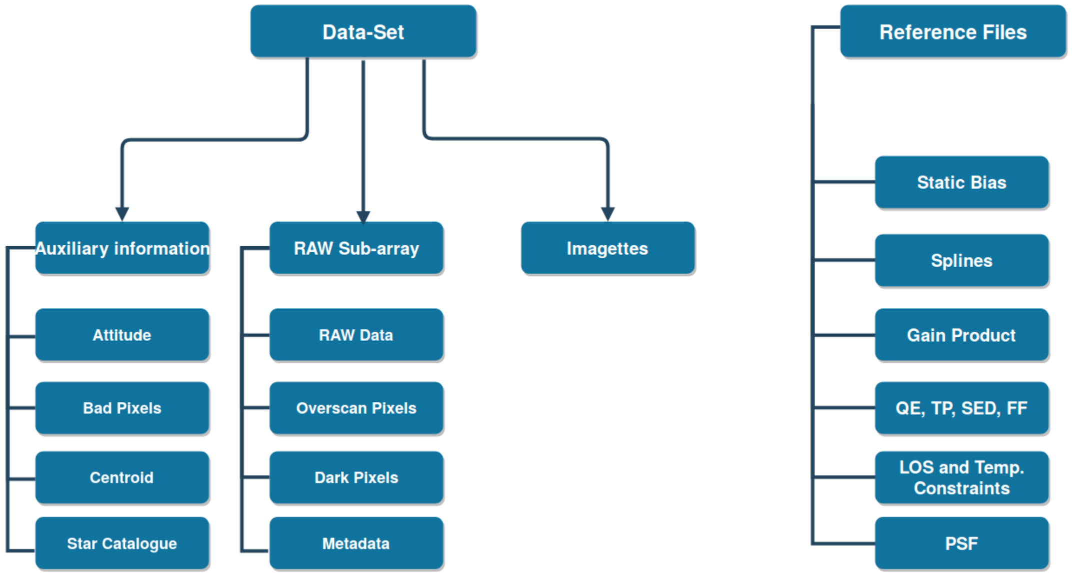


Figure 3.2: Scheme of the CHEOPS input and reference files data, with reference to the different products that are used throughout the reduction process.

- Bias and Read-out noise Correction
- Linearisation Correction
- ADU's - Photon Conversion
- Dark Current Correction
- Flat-Field Correction
- Event Flagging

Each of these submodules are explained hereafter:

3.2.1 Bias and Read-out noise Correction

The first sub-module in the calibration module of the CHEOPS data reduction pipeline is the bias and read-out noise detection and correction. As explained in the previous section, we can calculate the bias and the RON by analysing the bias distribution values.

Method:

The Cheops satellite won't have a shutter, so overscan pixels will be the option used in this mission for Bias/RON calculation. The Bias and RON quantification and correction process in the pipeline will occur as follows:

1. First, we extract the information of the overscan column's pixels, contained in the RAW FITS file.
2. For each exposure, we calculate the Bias and the RON. Bias typically follows a Gaussian distribution, where the Bias offset is the mean value and the RON the standard deviation, of the distribution. Therefore, in the pipeline, the Bias for each exposure is the mean of the overscan data for that exposure and the RON its standard deviation.
3. After, we will use a function of the auxiliary module dr_tools called smoothing, which smooths the bias values using a window filter. We smooth the Bias and RON values so we won't have spurious variations in Bias and RON.
4. To each exposure, we subtract the corresponding calculated Bias value, to each pixel of the RAW image sub-array.
5. In addition to this, a static Bias is also subtracted to the image. It is obtained during the ground calibration phase of the CCCD, by blocking external radiation and creating zero-second integration images. This Bias represents a static zero noise, that does not change much with time. This information is contained within a Bias frame, unlike the non-static Bias. It gives us information on small pixel-to-pixel non-uniformities that may affect the bias distribution.
6. Finally, we will bias-correct the top overscan pixels as they will be used later in the pipeline for the smearing and dark current correction sub-modules.

Results:

A static Bias frame is presented in figure 3.3. As can be seen, the static Bias is typically a small value between 0-1 ADUs, not contributing significantly to the total noise.

Additionally, a representation of the calculated bias and RON, for an exposure of a simulated data-set can be seen in figure 3.4. We represent the bias distribution for an exposure with 3 stacked images. In the top-right corner we show the mean value of that distribution, which will be equal to the bias offset of that exposure and the corresponding standard deviation which represents the RON. The Bias, for that exposure, is thus equal to 5261 ADUs and the RON to 10.36.

Since the RON is computed as the standard deviation of a normal distribution in order to calculate the RON per individual image, we divide the RON obtained for each exposure with the square-root of the number of stacked images (derived from a

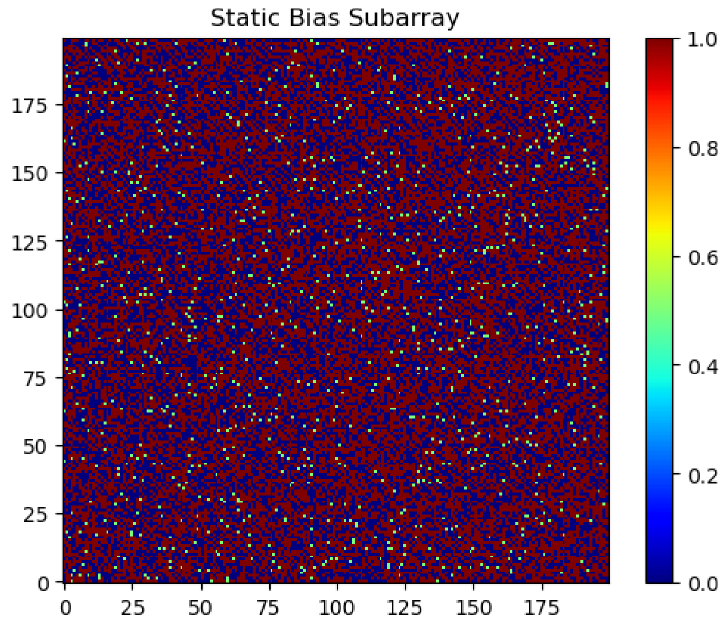


Figure 3.3: Static Bias Sub-Array. Obtained during ground calibration tests of the CCCD.

general standard deviation expression). In figure 3.4, we observe that the ”stacked” RON is around 10.38, so, for 3 stacked images, the RON in each individual image is equal to $\frac{10.36}{\sqrt{3}} \approx 6ADUs$. This represents a small value of RON being introduced in each individual sub-array, in each pixel.

The full bias correction process is seen in equation 3.1, where both bias are subtracted to the RAW Image.

$$\text{Image Unbias}_{exp,i,j} = \text{RAW Image Sub-array}_{exp,i,j} - < \text{Bias}_{exp} > -\text{Static Bias}_{i,j} \tag{3.1}$$

Thus, in order to unbias the pixel (i,j) in a certain exposure exp, we subtract to the sub-array's value the bias calculated for that exposure and the static bias value of the position i,j. It is important to note that the Static Bias$_{i,j}$ was obtained by subtracting the average bias value, to each of the static frame's pixels.

As previously explained, by stacking many images we will have, as a drawback, that the total read-noise will be higher than it would be if a single long exposure had been made. This source of error accumulates with the square root of the number of stacked images and it is independent of exposure time. Nonetheless, since nowadays the typical RON is very low, a higher RON may be preferred to having a saturated image.

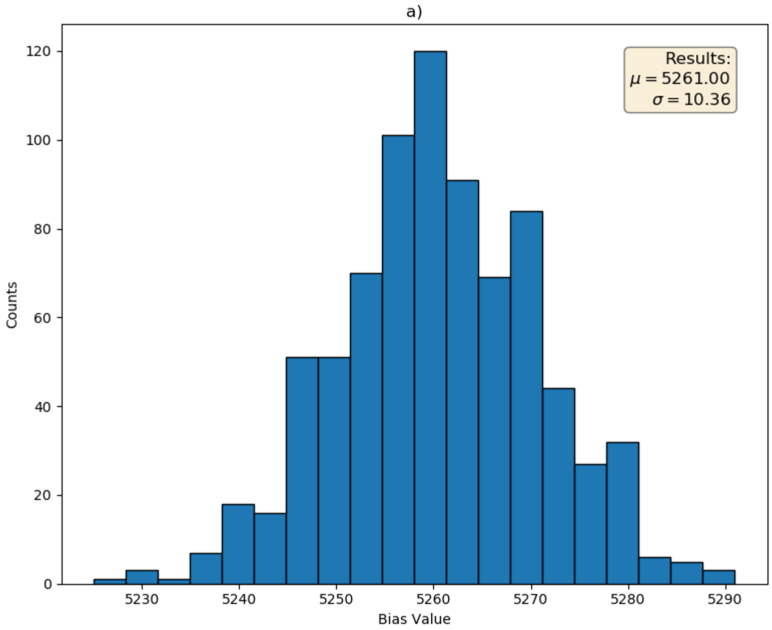


Figure 3.4: Individual Image Bias Distribution for a simulated exposure with **3** stacked images. In the top-right corner, it is represented the mean value of the distribution - bias offset, and the standard deviation - RON.

3.2.2 Non-Linearity Correction

This sub-module corrects for the non-linearity of the CCCD. As previously explained, non-linearity is measured as the difference between the CCD's response and a linear fit. In figure 3.5, we present the response of the CCCD during the calibration tests and the non-linearity results, taken from Deline et al. [2017]. In the left sub-figure we can see a plot of the mean value in ADUs vs the exposure time, with the straight line fit in red and the saturation level highlighted in dotted lines. The saturation level is seen as the point where there is a deviation of 3% to the linear fit. In the right sub-figure we present the non-linearity of each ADU level in percentile, in the range between 10-70% of the CCD's full well capacity. The non-linearity is obtained from the following formula:

$$\mathrm{NL}(\%) = \frac{\text{Read ADU} - \text{Linear Fit}}{\text{Read ADU}} * 100 \tag{3.2}$$

It can be noted that when the read ADU reaches lower values, the non-linearity amount increases steeply, the same happening when the read ADU values are higher.

Still, in the range represented in figure 3.5, non-linearity is always below 0.15%, in module, which is an acceptable value to have for this type of error.

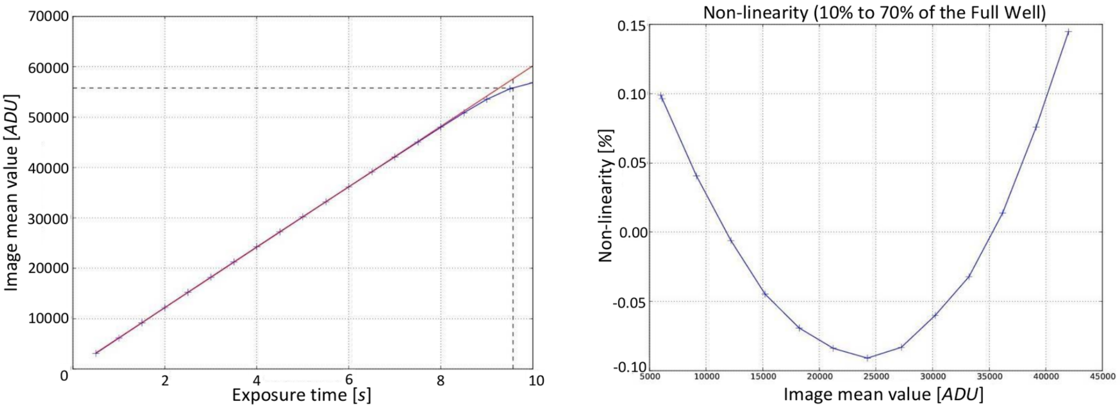


Figure 3.5: Non-Linearity tests and results. **Left:** Mean signal vs. exposure time curve with the straight line best fit (red) and the saturation level highlighted with the dotted lines. **Right:** Non-Linearity percentage vs the Mean Signal (ADU).

By having the information of the non-linearity percentage in each ADU level, we can create a so-called linearity curve, that plots the corrected ADU values vs the un-corrected ones. We can obtain this from:

$$\text{Corrected ADU} = (1 + \frac{NL(\%)}{100}) * \text{Non-corrected ADU} \tag{3.3}$$

In figure 3.6 we present the linearity curve of the CCCD, where we have in the x-axis a range of uncorrected read ADU values and in the y-axis the corresponding non-linearity corrected ADU. We present, for comparison, the curve for the case of a linear CCD. If we analyse the linearisation curve of the CCCD , we will notice how it starts to deviate more and more from the linear fit as the read ADU value goes to values higher than 50000 ADUs and, at a read ADU value of 58000 the CCCD saturates and the output ADU should be considered constant. As can be seen in this figure, the maximum corrected ADU value is a bit over 60000 ADUs. For comparison, for an ADC of 16 bits, the maximum representable number is 65535.

Method:

The points measured during the non-linearity tests can be interpolated by a set of cubic splines. These splines divide the linearisation curve in small sections and allow us to adjust and create the overall linearisation curve, present in figure 3.6. We are provided with a set of coefficients and boundary (knot) values that will permit us to perform non-linearity corrections. Each boundary determines the beginning of a

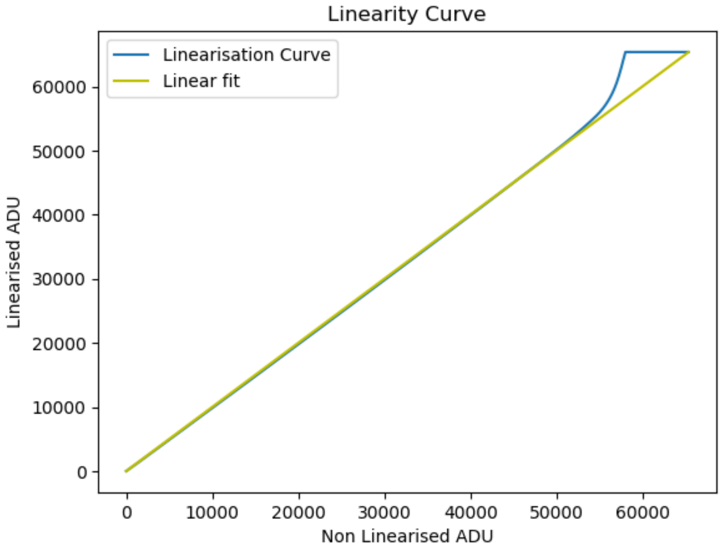


Figure 3.6: Linearisation curve of the CCCD, with a linear fit for comparison.

new spline and the end of the previous one. For a specific spline, the corresponding coefficients will determine how the curve will change with a variation in the read ADU value. The four existing coefficients will allow us to create a 3^{rd} order polynomial in that spline's range.

In order to correct an ADU value, it will be first assessed in which spline is this value found and the corresponding corrected ADU value will be equal to:

$$\text{Corrected ADU} = \text{poly}_{spline}[\text{ADU} - \text{Bound}_{spline}] \tag{3.4}$$

with,

$$\text{poly}_{spline}[x] = a_{spline}(x^3) + b_{spline}(x^2) + c_{spline}(x) + d_{spline} \tag{3.5}$$

where a,b,c,d are the 4 coefficients of that spline.

The Original Uniform Correction Method used inside the pipeline, as of the start of this work, to correct non-linearity in the unbiased data is the following:

- We first create a FITS table that has a column of non-corrected ADU values, ranging from 0 to $2^{\text{N bits}}$, and another column with the corresponding non-linearity corrected values, calculated using the splines method, explained previously.

- To correct each pixel value, we will get the closer ADU value in the non-corrected column to the one in need of correction, getting its table row index.

- We then extract the corresponding value (same row index) within the ADU-corrected column. This method was running much faster than previously implemented ones. The use of a correction table is more efficient than correcting each pixel one by one.
- To linearise the full data sub-array, we have to divide it by the number of stacked images and then linearise each pixel of this individual sub-array. We do this so to have an average pixel value of the individual image.
- When the linearisation process is finished, we multiply the sub-array again by the number of stacked images, getting a linearised stacked sub-array. We perform hence, an uniform linearisation.

We can conclude that it is best to try to use the CCD in the most linear region possible, as very low or very high ADU values are subject to higher non-linearity errors than intermediate ADU levels. We need therefore to control the exposure time of each individual image to maintain the read ADU values in this wanted range.

In section 4 we will go more in-depth with the topic of non-linearity.

3.2.3 ADU's - Photon Conversion

This next sub-module is the ADU-to-Photon conversion. This sub-module aims to convert the raw image data's units from ADU's to photo-electrons. As previously referred, the gain is a measure of the number of measured electrons necessary to have an output digital signal of 1 ADU. During the readout process, the amount of charge is first sampled into a voltage and then afterwards converted into a digital number. So, to have the information of the number of photo-electrons that were generated in the CCD's pixels, we need to reconvert the data from ADU units to electrons, therefore using the computed gain.

We can define a "good" gain value for a CCD detector. This value depends on the number of bits of the readout circuit's ADC as well as the full well capacity of the CCD. It represents an optimal value of the gain in order to have a good relation between sensitivity and electronic errors, therefore using the whole dynamic range of the CCD:

$$\text{Good Gain} = \frac{\text{Full well capacity}}{2^{N_{\text{bits}}}} \tag{3.6}$$

For the case of the CCCD we obtain a good gain value of 2.014 e-/ADU, for an ADC with 16 bits and the CCCD's full well capacity of 132000 e-/pixel.

The ADU-Photon conversion is made after the image bias correction, as it is imperative to correct the image for the bias offset, before converting to photo-electron units, otherwise we will just be propagating the bias offset error.

Method:

To obtain an image in photon-electrons we proceed in this way:

- Firstly, we extract the gain correction product, stored inside a FITS file, from the Reference_Files folder. The gain correction product contains information on the circuits reference voltages and the coefficients used to calculate the CCD's gain.
- Using this gain product, we compute the gain for each exposure.
- At last, we multiply the sub-array in ADUs with the calculated gain, in e-/ADU, for that exposure.

$$\text{Image}_{\text{photo-electrons}} = \text{Image}_{ADU,\text{exposure}} * \text{Gain}_{\text{exposure}} \tag{3.7}$$

Results:

In figure 3.7, we present the reader with the gain variation for a simulated data-set. As can be seen, the gain changes very little during each exposure - variations in the order of 10^{-7}. It is important to have a stable gain value during each measurement, since an unstable gain will create higher uncertainties in the charge measurement process. This gain variation is quite low and it does not reflect the gain uncertainty that will occur in real observations. The stability of the gain value is improved by keeping stable the CCD circuit's voltages and the temperature. The CCD's gain is dependent on these two factors, as was shown in the work of the ground calibration team. According to the Redbook [2013], we expect an error contribution of 5 ppm from gain variations, in real observations.

The simulated gain of the CCCD, calculated in the pipeline, is around 2.29 e-/ADU, as seen in figure 3.7. This gain value is quite close to the previously referred "good" gain value, which means that we will be using close to the whole dynamic range of the CCCD.

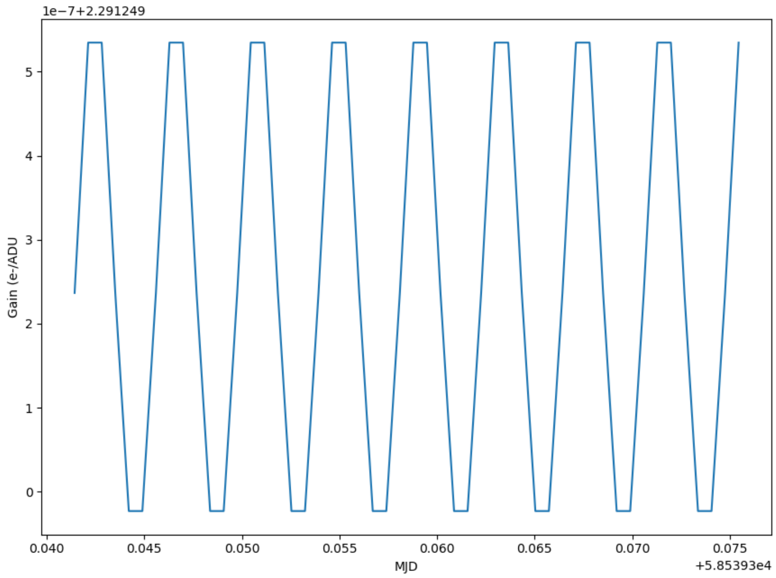


Figure 3.7: Gain Variation in an observation. In the x-axis we have represented the Modified Julian Date (MJD) of the time each exposure was taken, and its corresponding gain, in e-/ADU, in the y-axis. The plot does not show all the points of that observation for a better analysis on the variation.

3.2.4 Dark Current Correction

This next sub-module corrects for the existent dark current that affects the CCD's results. To do this, we use a set of pixels around the image frame that are shield from external radiation. Like this, we can estimate the dark current each time an observation is done. Dark Current is expected to change throughout the mission time-line. Towards the end of the mission we will have an higher dark current than at the beginning. This comes with the deterioration of the electronic components.

Method:

For calculating and correcting the varying dark current we need to take into consideration the accumulation and readout process of the CCD. This process takes into account that pixels which are readout later accumulate more dark current than the ones who are read first, as previously explained. Thus by creating a time map for the CCD frame, we will better estimate the amount of dark photon-electrons/sec/pixel. The dark current correction process in the pipeline follows the following steps:

1. We extract two different dark components (left and right) from the RAW FITS file.

2. We subtract the bias offset to the two dark columns/rows.

3. We perform the conversion of both dark pixel columns/rows from ADU's to photo-electrons, using the ADU to photon conversion sub-module. We can choose whether we want a non-linearity correction to these pixels.
4. We calculate the time map of the full array, explained below. We select the time map's pixels that correspond to the dark left and right columns.
5. We concatenate the dark pixel's information together in a single data array.
6. We average the number of dark electrons for each exposure and divide it by the average read time, obtaining a basic robust value for the dark rate of each exposure.
7. At last, we subtract to the sub-array of a certain exposure, the mean dark rate of that exposure times the time map for the sub-array's pixels, as shown in equation 3.8.

$$\text{Image Undark}_{exp,i,j} = \text{Image}_{exp,i,j} - (\text{mean dark rate}_{exp} * \text{time map}_{i,j}) \tag{3.8}$$

Results:

In figure 3.8 we show the time map array used for calculating the dark error. Values range from around 60 seconds, since it is the predefined observation time in order to obtain each exposure. The time map is calculated according to equation 3.9:

$$\begin{aligned} Time_{y,x} = 60 + \text{number stacked images} * (\text{Transfer time} + y * \text{CCD row read time} \\ + x * \text{pixel read time}) \end{aligned} \tag{3.9}$$

Where $Time_{y,x}$ is the time to read the pixel in row y, column x. Transfer time is the time taken for frame transferring from image to storage frame. ccd row read time is the time taken to read a row in the CCD and finally, pixel read time is calculated from $\frac{\text{CCD row read time}}{\text{number pixels per row}}$.

Figure 3.9 presents us with the dark rate distribution for a single exposure. We can observe that the average dark rate is, for this exposure, around 0.03 dark photon-electrons/pixel/sec. We can also observe a small number of dark rate values that are very high compared to the majority. These are spurious values that can appear, for example, in a situation where there were some small temperature variations during the readout process. The mean dark rate value is according to the theoretical expectation for the dark rate, back in figure 2.6, which should be between 0.01 and 0.1 dark photo-electrons/pixel/sec.

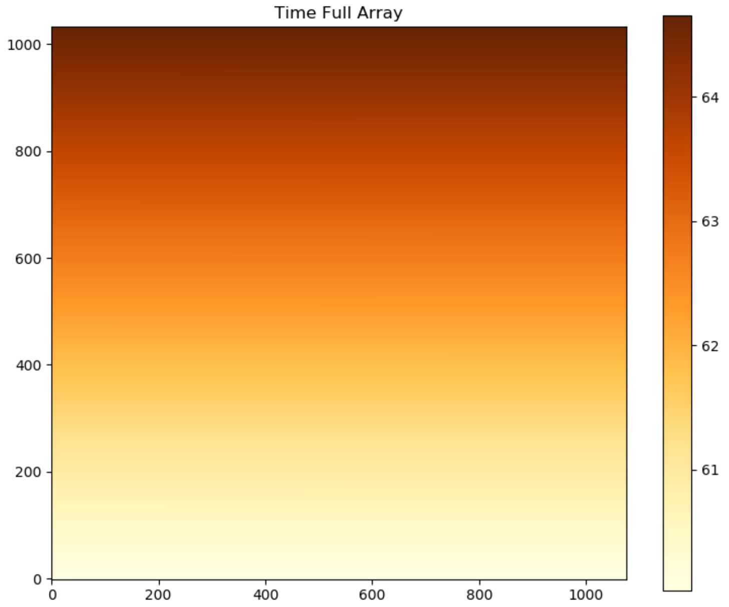


Figure 3.8: Time Map Full Frame, used to estimate the dark current in each pixel.

During an observation of 60 seconds, this dark rate value will translate in around 2 extra dark photon-electrons being measured in each pixel.

Therefore, the dark rate for the CCCD is quite low and it contributes only a small amount of error to the data.

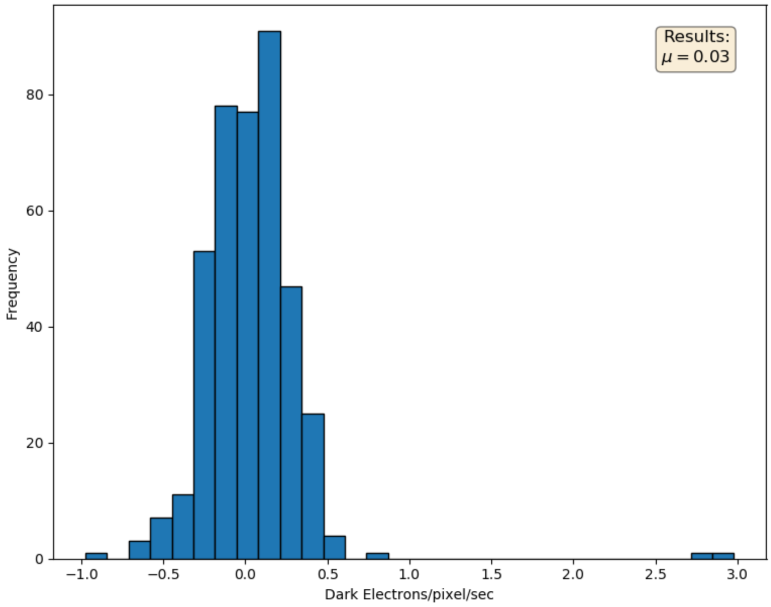


Figure 3.9: Dark Rate distribution for a single exposure. In the top-right corner we show the mean dark rate value for this exposure.

3.2.5 Flat-Field Correction

This next submodule is the flat-field correction. As previously explained, flat-fielding is an important process to correct for PRNUs. So we can perform the flat-fielding correction, we have contained, in the reference files folder, a number N of flat images taken at N different wavelengths and stored in a FITS file. Because we need to perform high-precision photometry, we are required to create a flat-field that considers the CCD's characteristics but also the target's.

To be able to do so in the pipeline, we need to have information on three different factors:

- The QE - It represents the percentage of incident photons in a detector that get converted to charge electrons. It is wavelength dependent.
- The TP - It is the percentage of photons which are not "lost" to phenomenon like reflection, diffraction, etc. Like the Quantum efficiency it changes with the incident light's wavelength. A telescope's mirror coating can alter the throughput - we can choose different coatings which are more responsive to certain wavelengths. The coating's choice should depend on what will be the observation's purpose.
- The SED - An object with a certain temperature emits light differently at different wavelengths. The SED represents the photon flux emission variation in function of the wavelength, for a given object's temperature. As an alternative to the SED, we may use the black body curve.

The QE of the CCCD can be analysed in figure 3.10, top sub-figure. From this sub-figure we can note that the CCCD has the highest quantum efficiency in the 450-650 nm range, which belongs to the visible part of the electromagnetic spectrum, in the blue range. The CHEOPS throughput is shown in figure 16, bottom sub-figure. The TP has its maximum response in the 550-650 nm range.

In figure 3.11, it is shown the SED for a star with temperature of 5170 K and its corresponding black body curve. For wavelengths $\leqslant$600 nm, the two curves are following each other closely, whereas they slightly deviate for higher wavelength values.

Method:

There will be three different ways to correct the data in this submodule. Each of these options are based on a method of attaching weights to each of the flat-field

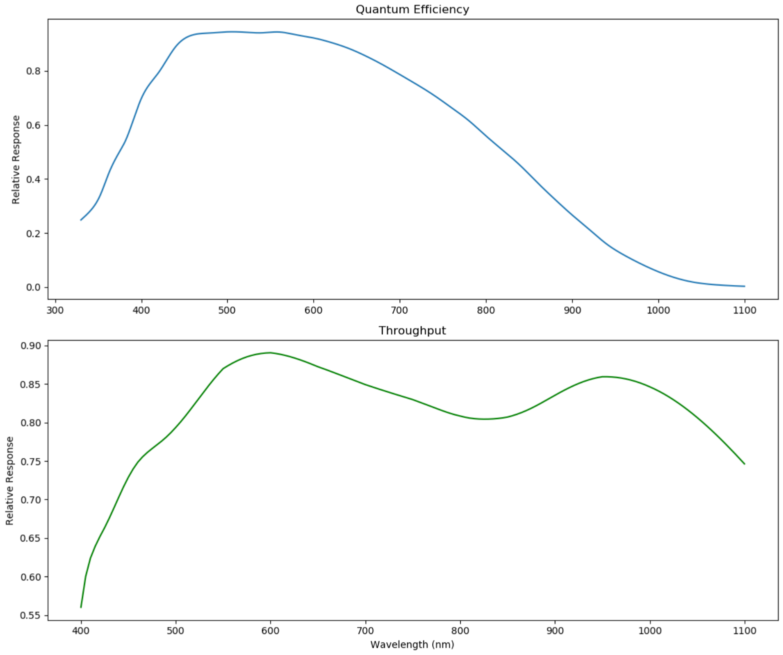


Figure 3.10: Quantum efficiency and Throughput of Cheops, for a range of wavelengths. **Top:** Quantum efficiency. **Bottom:** Throughput

images and, based on these weights, computing a ”master” flat-field. The options are:

- The use of a weighted flat-field, that uses the star’s SED to estimate the weight that each flat has. The star’s temperature information is stored in a star catalogue, kept inside a FITS file.

- Creation of an uniform master flat-field where we give the same weight to each flat image. This methods disregards any auxiliary information to calculate the final flat (QE,TP,SED).

- Finally we can use a so called standard flat, where we assume the temperature of the observed star to be the same as the Sun’s, calculating then the flat weights of each flat-field with this assumption.

We calculate the calibrated SED with the following formula:

$$\text{Calibrated SED} = QE * Throughput * \text{SED} \tag{3.10}$$

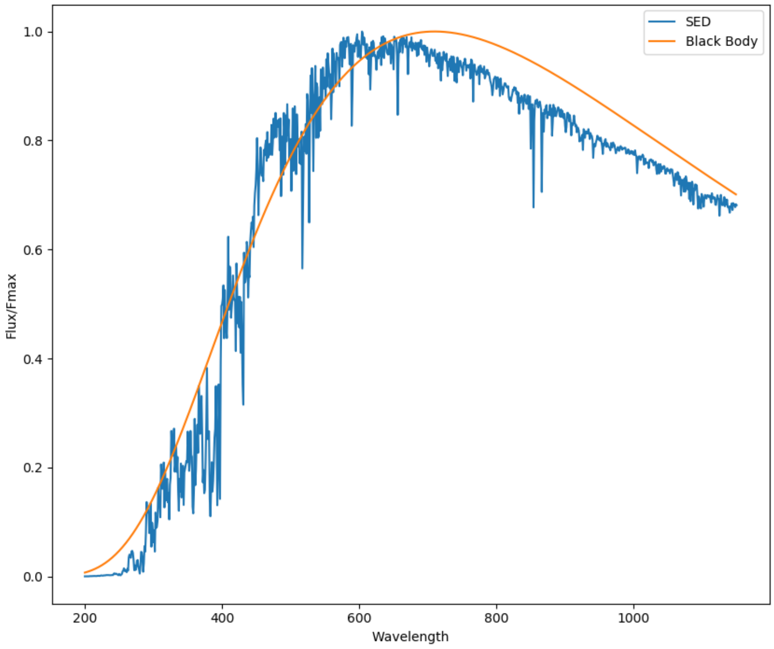


Figure 3.11: Flux Ratio of both the SED and Black Body curve for a star with T=5170K

and then, the flat weight of a certain monochromatic flat is given by:

$$\text{Flat Weight}_\lambda = \frac{\int_{\lambda-\lambda_0}^{\lambda+\lambda_0} f(\lambda)d\lambda}{\text{Max Weight}} \tag{3.11}$$

In this equation, λ represents the wavelength at which a certain flat-field was taken and λ_0 a wavelength bandwidth inputed by the user. $f(\lambda)$ represents an interpolation of a grid of calibrated SEDs and Max Weight represents the highest weight of all the flats.

As previously explained, each flat was taken at a different wavelength (monochromatic flat). We extract thus, the QE, as well as the throughput and the SED's flux ratio and calculate their response for a wavelength range. The weight of a certain flat is then obtained by multiplying these three factors and afterwards interpolating them for a given wavelength range.

To calculate the final flat-field, we use the next formula:

$$\text{Final Flat-Field} = \frac{\sum_\lambda^N Weight_\lambda * \frac{Flat_\lambda}{\overline{Flat_\lambda}}}{\sum_\lambda^N Weight_\lambda} \tag{3.12}$$

We can observe that the previously computed weights were used in the computation of the ”master” flat-field. At last we will correct each image by dividing it with the final (”master”) flat-field:

$$\text{Image Calibrated} = \frac{\text{Image}}{\text{Final Flat-Field}} \tag{3.13}$$

Results:

In figure 3.12 we present the reader with a comparison of the flat weights when computed in the three already mentioned methods: uniform, standard and weighted. We compute as well the flat weights using the Black Body Curve instead of the SED and present the results.

As can be observed, when using uniform weights the difference in the weights will be higher compared to the other methods as the CCCD’s, telescope’s and star’s characteristics won’t be accounted for in the flat calculation. The standard weights also deviate from the weighted ones, since we are not using the observed star’s real temperature. Finally, the black-body curve is a good approximation for the star’s emission, since the weights for this case are quite similar to the weights computed using the SED.

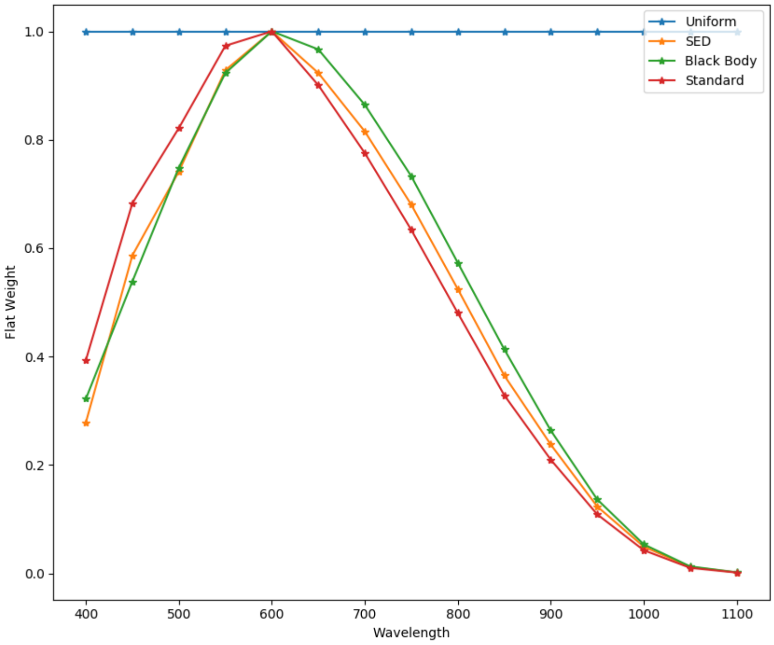


Figure 3.12: Flat Weights for a star with T=5170K, using different methods. In blue we have the weights using the uniform mode. In red the weights for a star with the temperature of the Sun (T=5777K) and using the SED. In green we present the weights calculated using the black body curve and in orange the SED. For this calculation we assume the real temperature of the star to be 5170K.

In figure 3.13 we show a final flat-field computed using the weighted flats option. In the top-right corner the mean value of the flat-field as well as the standard deviation are computed and shown. As expected, the mean value for the flat is 1, with a small deviation of approximately 0.004 around this value, which means that we have small non-uniformities in the CCCD. Although this flat is generally quite uniform, some of the flat's pixel show smaller responses (≈ 0.93).

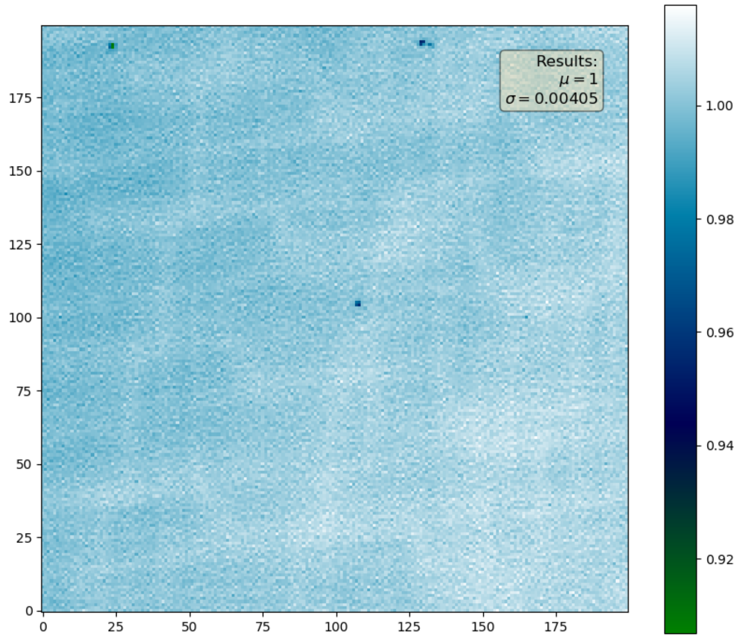


Figure 3.13: Final Flat-Field of the full array, computed using the weighted flats mode. The weights were calculated using the SED, for a star with T=5170K. The mean and standard deviation of the image are shown as well.

In conclusion, it was computed a flat-field which took into consideration the CCCD, the telescope's characteristics as well as the observed star's spectral type. For that, we assigned weights to each of the collected flat-field images in order to improve the creation of a final flat-field which would best reflect that specific observation. In the end of this sub-module, the image sub-array will be divided by the master flat-field sub-array, point-by-point.

3.2.6 Event Flagging

This next sub-module is called event flagging. Its goal is to analyse the collected auxiliary raw data of a specific observation and flag different events that may occur and are important to note.

This sub-module facilitates the study and analysis of collected data from the CHEOPS mission.

The evaluated parameters which are subject to flagging are the following:

- SAA events - If the observation was done in the region of the South Atlantic Anomaly a positive flag will be returned. In this region, the flux of energetic particles is higher than typical values, therefore this should be taken into consideration as the final results may be affected.

- Satellite's temperature - It comprises the temperature of the CCD and of the electronics components. High temperatures might affect the results, hence the importance of knowing the temperatures of each of these components during the observation and readout periods.

- LOS angle to the Sun - Light from the Sun can affect the obtained results. According to the Redbook [2013], the Sun must be outside the cone around the LOS of the telescope having a half-angle of 120°.

- LOS angle to Moon - Reflected light coming from the moon's surface might create unwanted signal in the data. The Moon must then not be inside a cone around the LOS of the telescope having a half-angle of 5°.

- LOS angle to Earth - The reflected light from the Sun in the Earth's atmosphere can compromise the results. The minimum angle allowed between the LOS and any illuminated part of the Earth limb, should be 35°. The earth can also occult the satellite to which a special flag is created for this case.

If, for example, during an observation, the temperature of the CCD was higher than a previously established threshold, this specific event will be flagged for that observation and a warning will be printed in the final report. An increase in the CCD's temperature will result, for example, in an increase of the dark current, which will consequently increase the total noise of the collected data.

All the obtained information will later be saved in a report file of that specific observation, which will be evaluated by a science team in order to determine the validity of that observation.

3.3 Data Reduction Correction

The goal of this module is to extract the already calibrated data and correct it for environmental effects. Like the Calibration module it is divided in sub-modules, which are:

- Smearing Correction
- Bad Pixel Correction
- Jitter Estimate
- Pixel to Physical Coordinate Conversion
- Background Estimate and Correction

In the following sub-sections we explain the different sub-modules present in the correction module of the Cheops pipeline.

3.3.1 Smearing Correction

The first sub-module of the Data Correction module is the smearing correction. Smearing is an effect that is more visible in brighter targets in which we can observe unwanted light trails from the top to the bottom part of the detector. This effect occurs in the CCCD during the frame transfer from the image frame to the storage frame, as the image frame is still exposed to light during this small period of time.
Method:

In order to do correct for this effect, we use the top ”overscan” rows, previously corrected for bias and converted to photo-electron units. The top overscan data is used since it collects the smear signal during the transfer. Using these overscan pixels, we compute the smear signal in each pixel.

We then subtract the calculated smear to the unsmeared image:

$$\text{Image Corrected}_{\text{exp,i,j}} = \text{Image}_{\text{exp,i,j}} - \text{Smear}_{\text{exp,i,j}} \tag{3.14}$$

Where $\text{Smear}_{\text{exp,i,j}}$ is the smear signal in exposure *exp* and pixel *i,j*. This is a process already used before in Quintana et al. [2010] for NASA’s Kepler mission.

3.3.2 Bad Pixel Correction

This next submodule is named the bad pixel correction. It is an important submodule of the pipeline and it identifies and tries to correct, as much as possible, the collected data for bad pixels, which affect the results.

There are four types of bad pixels which need to be identified and corrected for:

- Cosmic Ray Hits
- Hot
- Dead
- Telegraphic

Cosmic Rays (CRs) are high energy particles, mostly protons, who come mostly from stellar winds. Since the Cheops satellite is in orbit it is subject to the incidence of more CR than it would at the Earth's surface. This makes the need to correct for this source of error of high importance. During an observation it can happen that a cosmic ray will deposit its energy in the CCD causing unwanted signals being created in some of the CCD's pixels. Typically a CR's effect on a CCD is characterised by a flux overflow on a group of neighbouring pixels.

Hot Pixels are pixels who are always above their neighbours. In contrast, dead pixels are always below. Telegraphic or crazy pixels are the most noisy pixels in the array detector.

Method:

In order to detect the different groups of bad pixels, some methods are employed :

CRs are described as spurious outliers in time. Therefore, we use a sharpening filter to obtain an enhanced image and afterwards compute the residuals by the temporal mean. Afterwards, we examine the residuals for outliers and compare with a robust deviation of the residuals. Similar methods were used in Samadi et al. [2007], Jenkins et al. [2010], and also van Dokkum [2001]. In order to improve the CR detection, we also apply this method to the imagettes, which are smaller sub-arrays of 30*30 pixels, centered in the observed target. Imagettes improve the detection of CRs due to their lower exposure time (high cadence). An example of an imagette is presented in figure 3.14.

Afterwards, we merge the two bad pixel arrays, one computed using the sub-array, the other with the use of the imagettes.

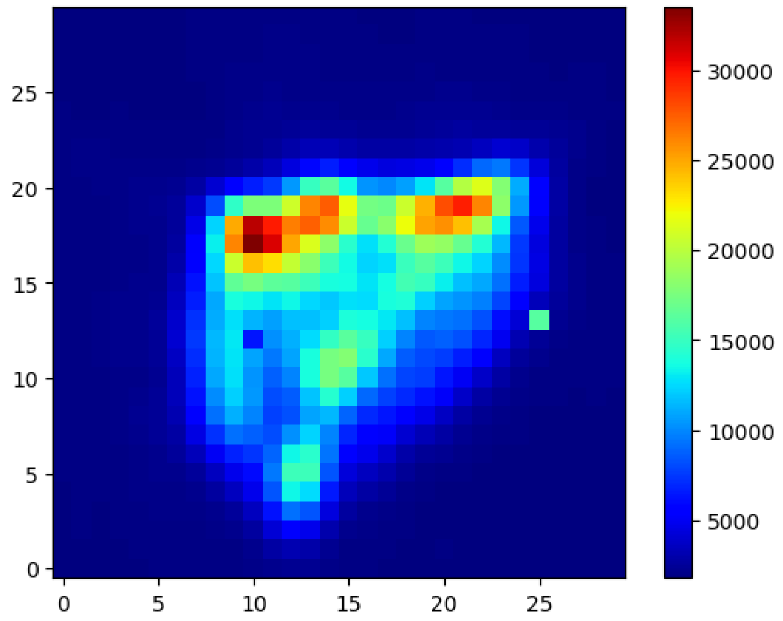


Figure 3.14: An imagette (30*30 pix) of an exposure. We can see that it contains the pixels with higher variations of the sub-array.

For the Hot/Dead pixels, we find the ones whose temporal mean value is extremely above / below the robust deviation of itself. Finally, for the telegraphic pixels, we search for the ones whose temporal deviation is lower than a threshold calculated from the deviation of the variation distribution. Since these last three types of bad pixels are not supposed to change much through time, their pixel information and location is stored in a FITS file, kept in the Reference Files Folder.

The idea behind the correction process of these bad pixels is to average the bad pixel using the neighbour's values with the use of a median filter. In the case of CR pixels, we have to also make sure that the flux in that pixel did not increase after the correction took place. If this occurred, the pixel's value will be kept.

3.3.3 Jitter Estimate

This next sub-module is called the jitter estimate. Throughout each exposure and observation, the star's centroid position is expected to change in the sub-array due to the CHEOPS spacecraft's pointing jitter. The spacecraft rotates around its orbit, in order to keep the satellite's radiators opposite to the Earth, for thermal stability. This creates pointing uncertainty when observing - jitter. This effect would still be present even if this rotation would not take place, although in a smaller magnitude.

A precise calculation of the star's position in the sub-array is crucial, in order to later define a good aperture in the photometric extraction module (subsection 3.4). If not considered, situations where some of the pixels that contain the star's flux are outside of the aperture may occur, creating oscillations in the final light-curve. These oscillations, if not considered, will create a well-noticed dependency of the flux with the spacecraft's roll angle.

Method:

The method used in the pipeline to determine the Center of Gravity of the image is based on the work of Vyas et al. [2009]. In this article there are presented some methods of calculating the center of gravity of an image, and in this case, the one used in the pipeline is the Iteratively Weighted Center of Gravity.

As the name suggests this is an iterative method that uses a weight function, given by a 2-D Gaussian and a Scipy package's module called center of mass, which tries to reach a convergence in the residuals value of the centroid's estimation.

The IWCoG is calculated, at each iteration by:

$$X_c, Y_c = \frac{\sum_{i,j} X_{i,j} I_{i,j} W_{i,j}}{\sum_{i,j} I_{i,j} W_{i,j}} \tag{3.15}$$

Where X_c, Y_c are the x and y coordinates. $W_{i,j}$ is the weight, given by the Gaussian, $I_{i,j}$ is the image's intensity at that coordinate point and $X_{i,j}$ the pixel position in the grid.

We start the method with an assumed center at the position (100,100). In each iteration, we analyse the spread between the center of the previous iteration and the computed center from the center of mass function and check whether it is smaller than an inputed tolerance. If this holds true, the function breaks and the centroid will be the one computed in that iteration.

3.3.4 Pixel to Physical Coordinate Conversion

This next's sub-module's goal is to, using the information contained in the metadata of the sub-array and the centroid's position computed in the previous sub-module, calculate the physical coordinates of each pixel's center for example, the right ascension and declination. The new coordinates will be given in terms of the FITS World Coordinate System (WCS).

The calculated information will be stored in a FITS file, for easiness of access.

3.3.5 Background Estimate and Correction

This next sub-module aims to estimate the rate of background photons that the data is subject to. As previously explained, each exposure may be contaminated by light that comes from other star's in the field of view, or even stray-light coming from the Sun and reflected by the Earth's or Moon's surface or even directly from the Sun itself. This light increases the overall photon flux in the collected data and needs to be estimated as it is an important noise source.

Method:

In order to make this estimate in the pipeline, we evaluate the sub-array's pixels which are contained in a donut-like structure, centered in the (100,100) position in the sub-array, with an inner radius of 30 pixels and an outer radius of 100 pixels. Afterwards, we create an histogram of these pixel's values and smooth them, with a low-pass filter. At last, we will fit this histogram's values with a Gaussian fit.

We assume the mean background photon rate as the mean value of the fitted histogram. In the end, we subtract the mean background rate times the exposure time to all the sub-array so to correct for background photons' effect. This procedure was based on the work of Drummond et al. [2008].

$$\text{Image Background Corrected}_{exp} = \text{Image}_{exp} - \text{Back}_{exp} * \text{exp. time} \tag{3.16}$$

where Back_{exp} is the mean background level of that exposure and exp.time is the exposure time (=60 seconds).

3.4 Photometric Extraction

The last module in the Data Reduction Pipeline is the Photometric Extraction. This module aims to estimate the contamination present in the data and extracts the light curve and analyses its quality.

It is also divided in many sub-modules:

- Simulation and Contamination Estimate

- Aperture Photometry
- Light-Curve Quality Analysis

3.4.1 Simulation and Contamination Estimate

This next sub-module aims to estimate the amount of flux coming from other stars in the field-of-view of the target star, that may affect the aperture flux measurements. We use a star catalogue that contains the position of these stars in the sub-array, and, using the PSF, estimate the flux that is coming from these nearby stars and, separately, from the target star.

The contamination is measured as:

$$\text{Contamination} = \frac{\text{Sim. Flux Background}}{\text{Sim. Flux Target}} \tag{3.17}$$

This method was based on the work of Bordé et al. [2010] for the COROT space mission.

3.4.2 Aperture Photometry

This sub-module's objective is to extract the photon flux contained in each exposure.

Method:

To do so, we use the dr_tools module's function disc, and create a circular aperture whose center is that exposure's centroid position, calculated previously in the jitter estimate sub-module. We create N different circular apertures whose centers coincide with the center of gravity of each exposure. The circular aperture's radius is manually introduced and it has a default value of 33 pixels. This aperture value should always be at least twice the size of the PSF's radius, according to Aigrain et al. [2015].

To obtain the flux we sum the amount of photons present inside the aperture. The pixels who fall in the border of the aperture are counted as a percentage of the pixel which falls inside an aperture, so if a pixel is half inside the aperture its counted value will be multiplied by 0.5.

An alternative method is based on the approach of Bryson et al. [2010] and Auvergne et al. [2009]. Using the simulated flux from the target and the nearby star's, computed using the PSF in the previous sub-module, we find an aperture radius that minimizes the value of a Noise-to-Signal equation, which is an inverted version of the expression presented in subsection 2.3.3. In this equation we use not only the simulated fluxes but also the RON calculated previously in the pipeline. Lastly, we computed the aperture flux with this new optimised aperture.

Results:

In figure 3.15 we can observe in the left sub-figure the created photometric mask from which we extract the photon flux for that exposure. The circular aperture is centered on the computed center of gravity of that exposure. In the right sub-figure we can analyse the flux variation for that observation, where we have the flux in the y-axis and the Modified Julian Date (MJD) in the x-axis.

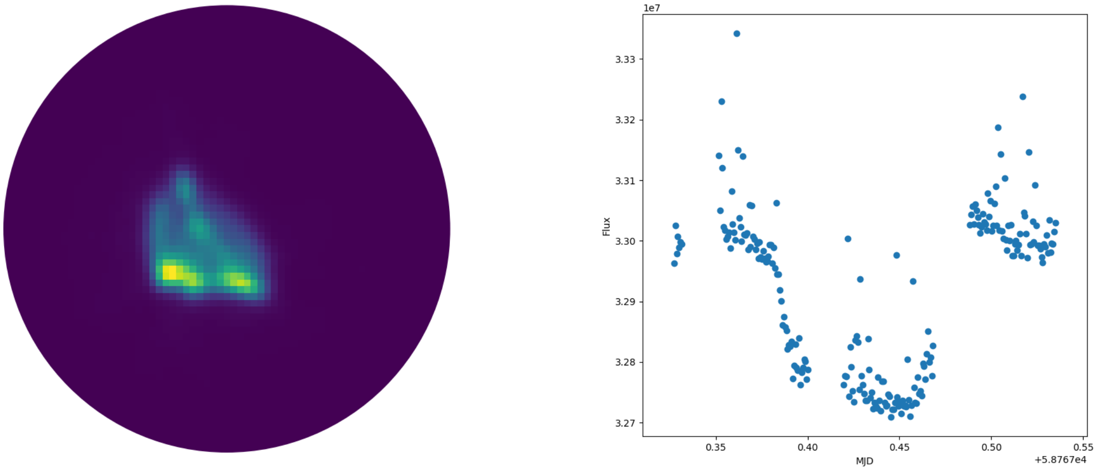


Figure 3.15: The output of the aperture photometry module. **Left:** The circular aperture created in the pipeline, centered in the target's computed center of gravity, for a single exposure. **Right:** The flux variation vs observation time, i.e the final light curve.

3.4.3 Light Curve Quality Analysis

Lastly, the last sub-module of the photometric extraction module is the light curve quality analysis. As the name suggests, the goal of this sub-module is to evaluate the produced light curve.

Method:

The quality of the light curve is quantified by a set of parameters:

- Median

- Robust Mean (Hoaglin et al. [1983])
- Robust Deviation
- Point-to-Point Precision (P2P)
- Quasi-Combined Differential Photometric Precision (quasi-CDPP), Aigrain et al., 2015 and Christiansen et al. [2012]

The first three parameters are quite straight-forward to understand. The P2P is the median absolute deviation on the point-to-point difference of the normalized flux. The CDPP is essentially the depth of the transit in parts-per-million (ppm) when the SNR is equal to one (lower CDPP will mean a better precision). Quoting Aigrain et al. [2015], the quasi-CDPP is the median of the standard deviation of the light curve evaluated in a moving window of a given duration. The quasi-CDPP is evaluated in different time windows of 5 min,30 min 2.5h, 6 h, 6.5 h and 12h. If the observation duration is smaller than the window time, the quasi-CDPP will be zero.

3.5 Bits Reduction

It was tested, as a case study, the effect of removing up to three bits of information from each pixel value, in the RAW Sub-array. For a 16 bit ADC, by removing information of up to 3 least significant bits (lsb), we would be removing up to 7 ADU's from each read pixel. This was tested due to the fact that, as mentioned before, there are many constraints in this mission, therefore, by sending less data each observation, from the satellite to the ground servers, the total cost will be smaller, and more data can be sent in the same amount of time.

Method:

We created a python script that would reduce up to 3 least significant bits of information to the RAW Sub-Array(Data + Overscan+ Dark) and ran this "reduced" sub-array in the pipeline, evaluating the obtained results. To assess the results, the quality of the produced light curve was analysed and compared to the non-reduced case. We also analysed the outputs of all the other sub-modules to understand if they would provide similar results.

Results:

When taking 3 bits of information to each pixel, it could be seen just a very small reduction in the quality of the produced light curve compared to the non-reduced case. On the other hand, we noticed that the calculated Bias and RON, for an exposure, would slightly change due to the reduction, as can be seen in figure 3.16. In this figure, we present on the left sub-figure the Bias and RON for a normal sub-array without reduction, and on the right the 3 bits reduced sub-array. We can see that the measured mean value (Bias) and the standard deviation (RON) change from the non-reduced to the reduced case. Also, the bias distribution stops having a continuous profile, and we can see higher variations in the obtained bias values, stemming from the reduction. Although the Bias/RON change may seem small, they represent an addition of error to the final results, and an inability to correctly quantify both the RON and the Bias. In addition, by reducing bits, we were later unable to correctly smooth the BIAS distribution of each observation.

Fortunately, the latest version of the pipeline will have a reduced mode where the Bias and RON are variables computed on-board and stored inside the RAW sub-array. Therefore, we will not lose this information even if we reduce the RAW data by a certain number of bits, as these stored variables will remain unchanged.

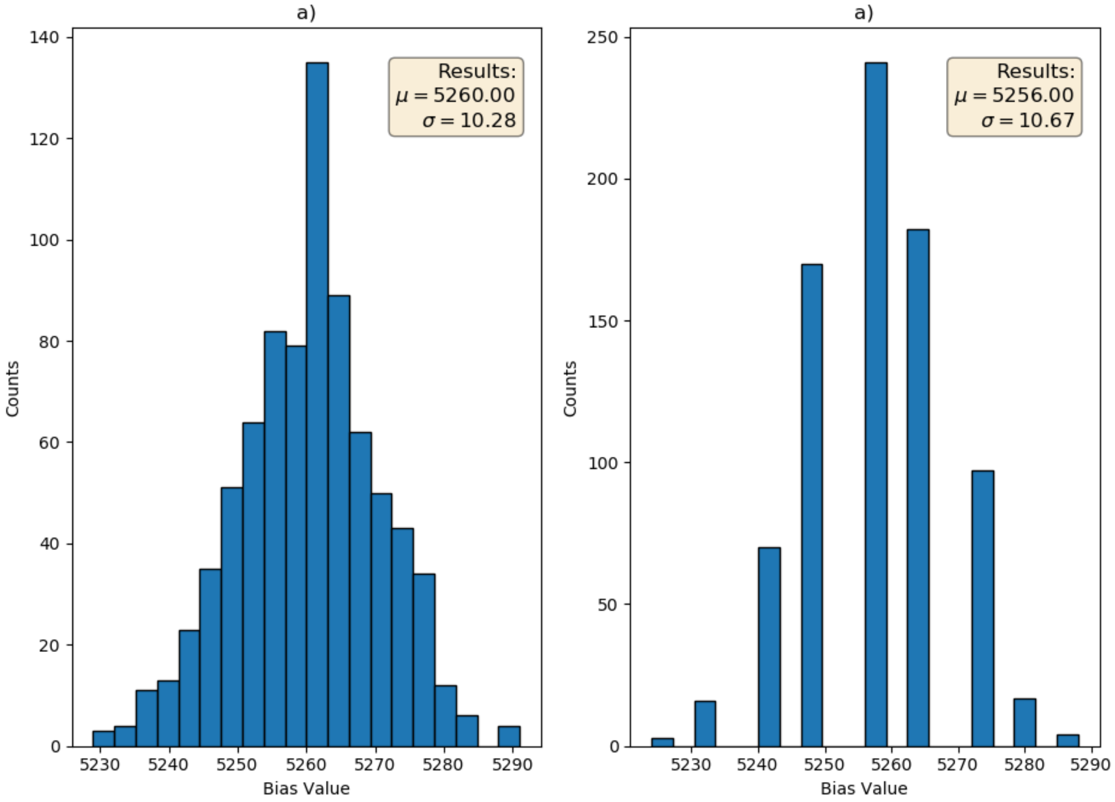


Figure 3.16: Bias Distribution of an exposure, with 3 stacked images, for both the normal and reduced cases. **Left:** Normal case without bits reduction. **Right:** Sub-array reduced by 3 bits.

4

Pipeline Modifications

Introduction:

As previously explained, one of the goals of this work was to replicate the original CHEOPS DRP, creating a simpler version and developing improved algorithms. This implied not only getting familiarised with this software but also with commonly used file formats and packages. In appendix A.1 we present both the more important file formats and Python packages that were used throughout this work. In addition, in appendix A.2 we make an overview of the replicated DRP, explaining how it was built and what are its outputs.

Looking to the calibration module, the linearisation correction sub-module was the one targeted for improvements. The reason for this was that the original uniform correction method, that was based on an ADU pixel by pixel average method is insufficient to provide accurate non-linearity corrections in the CHEOPS data, particularly for the typical cases where the CHEOPS image observations are stacked, losing this way the full information that we would have in case of individual images. This is specially more problematic when we have significant target centroid variations since the original correction method fails to account unstacked-to-unstacked image changes in the target's observation.

The target star's position is supposed to change during observation up to a small number of pixels. Therefore, if the number of photons in a certain pixel varies significantly from unstacked-to-unstacked image, this variance won't be accounted for using the uniform linearisation methodology present in the original uniform correction method.

In the next subsection a new linearisation method is presented to the reader.

4.1 Improving the Linearisation module - Linearisation with Imagettes

We propose an alternative method which will benefit from the use of the information contained in the imagettes. For a single sub-array image, stacked by N individual exposures, we will have the correspondent N unstacked imagettes. All this information is stored in the respective FITS file in the pipeline.

These imagettes were originally designed for cosmic ray detection and possible correction of the stacked sub-array images.

In this new method we propose the correction of non-linearity errors using each one of these imagettes, for the pixels where we have full individual information. The use of these imagettes is relevant since it contains the information of the majority of the photons that will be inside our photometric mask. It is also in these central parts of the Point Spread Function (PSF) where we have higher pixel-to-pixel variations which will become important for the non-linearity correction.

This new method will work, step-by-step, in this fashion:

1. For each observation, we acquire all the imagettes of each exposure.
2. We will extract the centroid information of each imagette, already present in the imagette's FITS file.
3. We want to linearize the full image, so, for the pixels for which we don't have the full information included in the imagettes, we use the simpler method.
4. We unbias each imagette, using the calculated bias.
5. For each exposure, using the centroid information, we will assess how many overlapping pixels, in both axis, there are in the imagettes - we can only accurately linearise these pixels. An example of this can be seen in figure 4.1.
6. We linearise each of these reframed arrays, stacking them afterwards and changing the newly linearised pixels in the linearised sub-array.

Problems:

We encountered some problems in the development of new linearisation methods, mostly due to the treatment given to low and to negative ADU values in the splines/correction table methods.

For low read ADU values (<100), we would get a corrected ADU value much higher than should be expected, overall increasing the amount of error we would have in the computed flux. This problem can be seen in figure 4.2, where there is an offset when the read value is 0 ADUs. This is included in the currently available linearization correction product. We expect this offset to be corrected in the future releases of the linearization product.

For these values it can be seen that the output ADU will be significantly higher than their read counterpart.

To improve this problem with the linearization curve, we have adopted the strategy that is currently implemented in CHEOPSim, which is to subtract this offset to the full curve. Additionally, negative values, that could be present, were not linearised, as they should just be accounted as a consequence of the RON and respective bias correction.

To demonstrate the performance of this new alternative correction for non-linearity we designed a test module which is explained in the following sub-section.

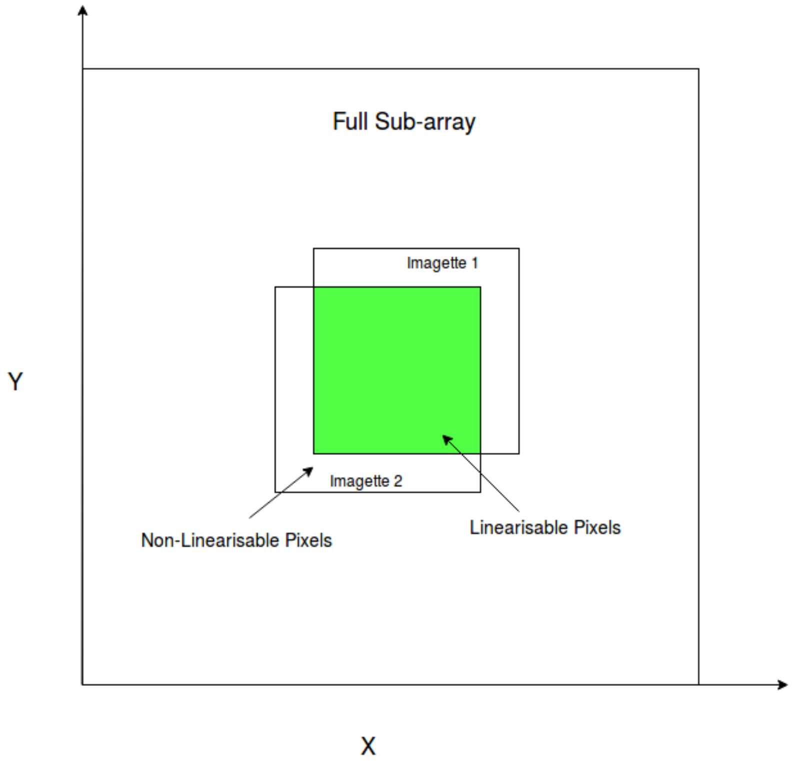


Figure 4.1: Diagram of the imagette linearisation module. Since each imagette may offset from each other by a small number of pixels, we will only linearise the overlapping pixels, represented in green. Image is not to scale, and it serves as representation only.

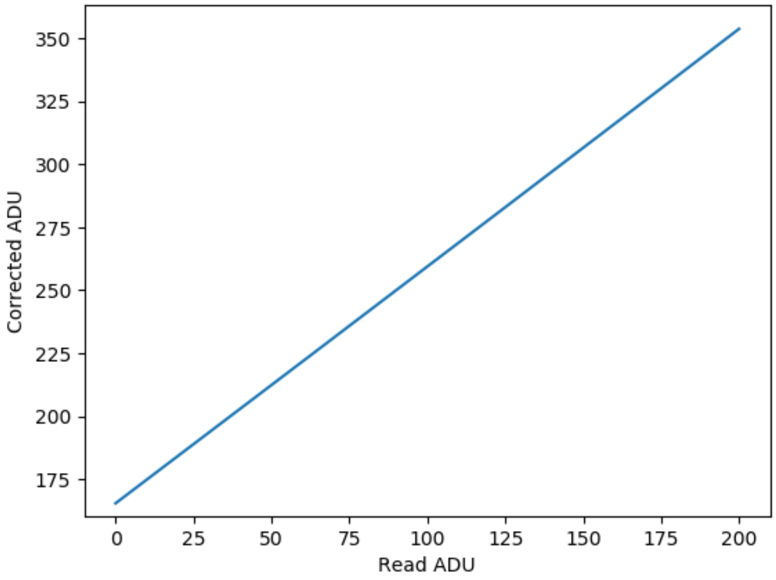


Figure 4.2: Linearisation Correction Curve for low read ADU values. We can observe the existing zero-point offset

4.2 Linearisation Functional Test

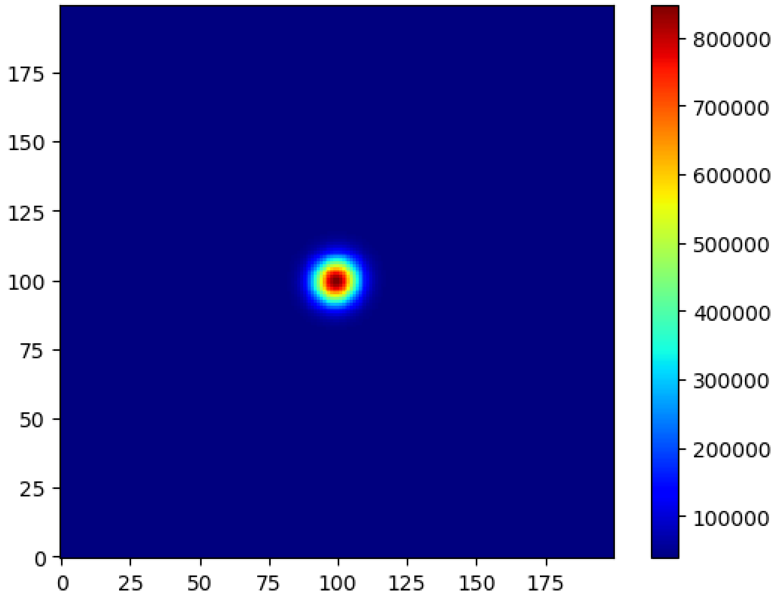


Figure 4.3: Simulated Stacked Truth Data Frame. The Star is simulated as a Gaussian with a variable centroid, amplitude and width, inputed by the user.

In this section we describe a functional test used to check and compare both linearization methods. In order to do this, we needed to create a simulation reproducing as best as possible the main problems that we expected to encounter in real observations more specifically, the ones relevant for this submodule.

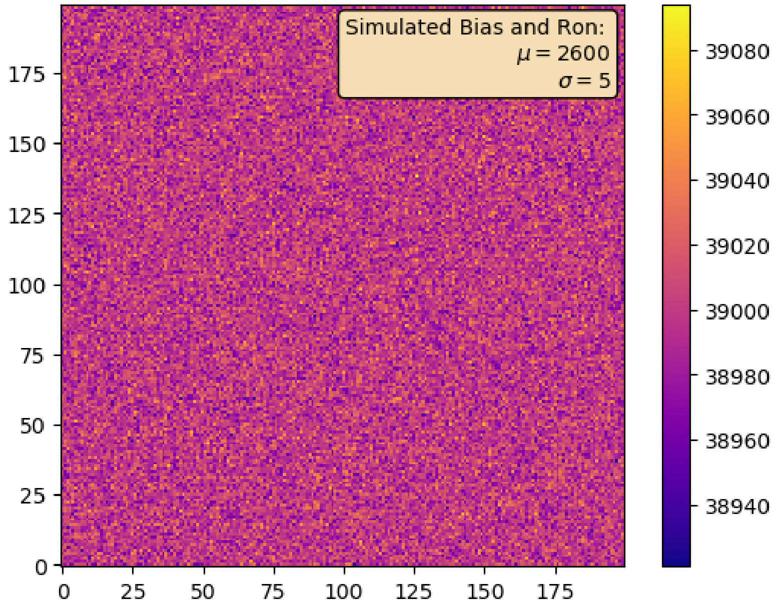


Figure 4.4: Simulated Stacked Bias Frame, using a random normal distribution. The simulation was made with **15** stacked-images. On the top right corner we have the individual image's user inputed bias and RON.

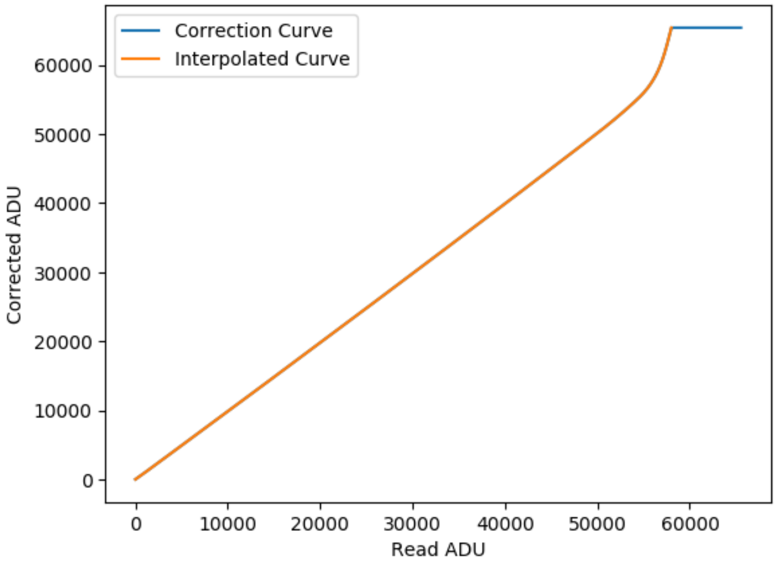


Figure 4.5: Linearity Correction Curve and Interpolated Curve, used for adding non-linearity

The simulation and testing process worked, step-by-step, as follows:

1. A star was simulated by generating a 2-D Gaussian, with an user inputed amplitude and width. The Gaussian's center was defined by a normal distribution , in both axis, where the mean value is the sub-array's center (100) and the standard deviation an inputed value (usually around 2-3 pixels, named centroid's sigma for simplicity). This variation intended to replicate the change in the target's center that will be observed in a real observational situation. An example of a simulated star is present in figure 4.3.
2. We added a non-linearity error point-by-point. For performance purposes, we created an inverse correction table, interpolated from the original correction table. The inverse curve is plotted in orange in figure 4.5.
3. We added a bias and RON to the image. The bias level was also user inputed, with a standard deviation representing the RON. A simulated bias frame can be seen in figure 4.4. The bias will then be added pixel by pixel. The standard Bias and RON values are the same as in figure 4.4, with a bias of 2600 ADUs and a RON of 5.
4. We extracted the imagettes from each individual image and their corresponding centroid information. This information was obtained from having previously stored the centroid's position in step 2. For each exposure, the centroid's position is the mean of the individual images' centroids.
5. We generated N exposures by summing M individual images. The N and M values vary depending on each simulation.
6. The images are then bias corrected using the average value for the bias, keeping this way the RON, simulating what will happen with real data. This correction was done in the same way as in the original DRP.

4.2.1 Test Validation Criteria

Here we describe how we check and compare both linearisation methods.

In short, we simulated data and added three possible error sources:

- Pixel non-linearity
- Centroid variation
- Bias and RON

The differences created by these three different error sources can be analysed in figure 4.6. The y-axis represents, in each sub-figure, the difference, pixel by pixel between the expected flux result generated by the simulated image and the flux of the corrected product data, for the pixels of the used aperture. The error differences were normalised using the number of stacked images, to have a mean error per individual image. In each sub-figure, the points which belong to the imagette area are plotted in orange.

- In sub-figure **a**), we present the flux differences for a simulation with all three previously referred error sources present. The standard individual values for the error sources: Bias, RON and centroid were: 2600 ADUs, 5 ADUs and 1.5 pixels.

- Conversely, sub-figure **b**) is the result of a simulation without centroid variation.

- Additionally, sub-figure **c**) represents a simulation with pixel non-linearity as the only source of error

- Sub-figure **d**) a simulation was made without bias/RON.

For all of these plots we have, as well, the information on the mean pixel-to-pixel difference and the RON, when applicable.

We can observe that **a**) has the highest point-to-point differences due to all the errors being combined. In **b**), the error created by the RON is highly prominent, as expected for a simulation of 15 stacked images with an individual RON of 5. As seen before, RON increases with the square-root of the number of stacked images, therefore we have high flux differences in the results, even after we correct for the bias offset.

As can be seen in sub-figure **c**), the error present is caused from the correction curve interpolation since there are (very) small differences between both curves. This type of error is quite small, compared to the other two, as we can see from the results. On the other hand, in **d**) we can notice the existence of some drifts, for lower ADU values, caused by centroid variations and uniformised pixels.

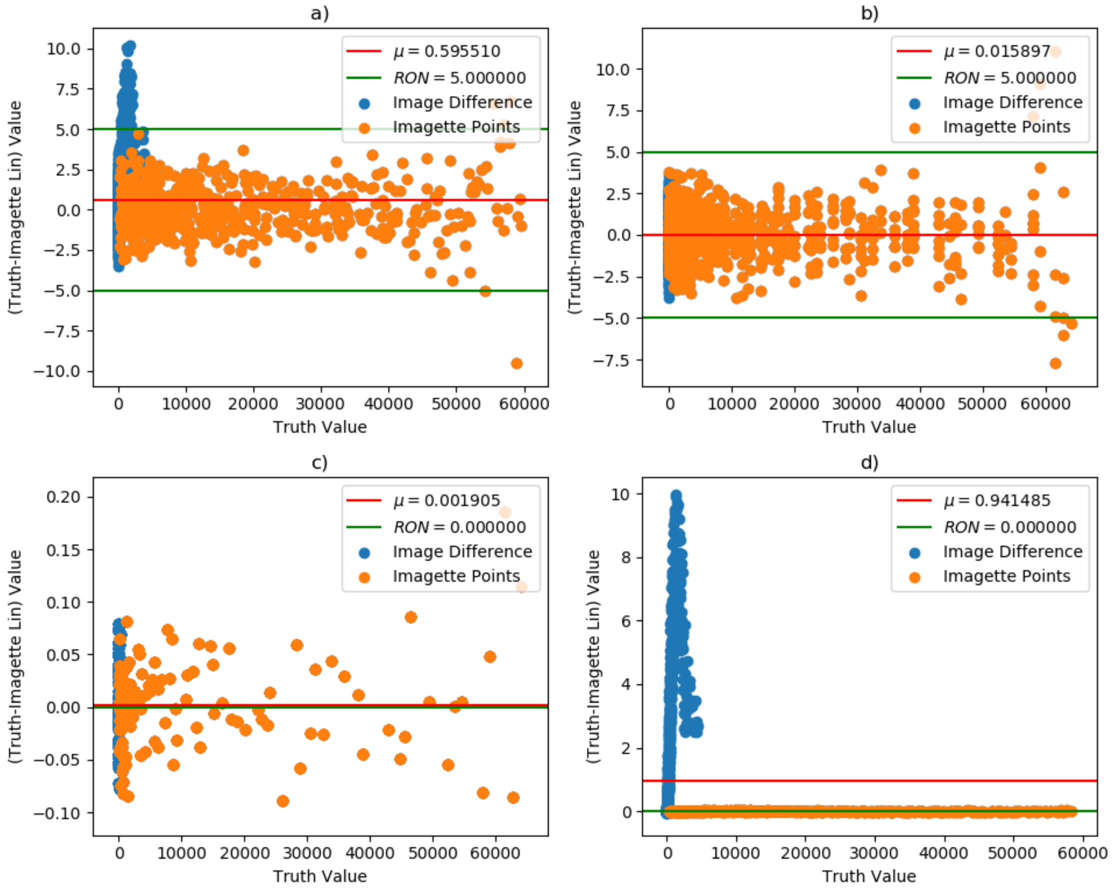


Figure 4.6: Point-to-Point difference between a simulated,error-free sub-array and a imagette linearisation corrected image obtained by stacking **15** individual images. The results represent only the aperture pixels, and the values are uniformised - average error per unstacked image. The values in orange represent the pixels inside the Imagette area. The green horizontal line represents the RON and the red line (μ) represents the average point-to-point difference. **a**) Plot with a centroid variation σ of 1.5 pixels. **b**) No centroid variation. **c**) No centroid variation and no RON. **d**) No RON and centroid variation σ of 1.5 pixels.

Centroid's Variation Results:

In order to understand the centroid's effect in the linearisation methods, we analysed the differences between the truth sub-array and the imagette linearised sub-array. These differences can be seen in figure 4.7, for a 15-stacked image simulation without RON and with a centroid variation following a Gaussian distribution, with a mean in the sub-array's center (100,100). The results shown are for the aperture pixels only.

We note that there are some differences between the two sub-arrays, inside the aperture pixels but outside the imagette area (rectangular area in the center). These differences occur because we perform an approximation (above or below the real value) when we linearise the aperture pixels uniformly. Since our gaussians were simulated with a width of 5 pixels (σ of the Gaussian), in each axis, we will still have big variations, from image-to-image, in the region outside the imagettes area, which won't be accounted for. The results in figure 4.6, sub-figure **d**) can be related with the problem we see in figure 4.7, since they represent the same error for low ADU values. The Gaussian's highest ADU values fall inside the imagette area so they will be properly corrected with the imagettes. The same will not happen for lower ADU values who will fall outside the imagette's area, but still inside the aperture. We can conclude this from seeing that, in sub-figure **d**), the pixels inside the imagette will have close to zero point-to-point differences and that the highest disparities occur for non-imagette pixels.

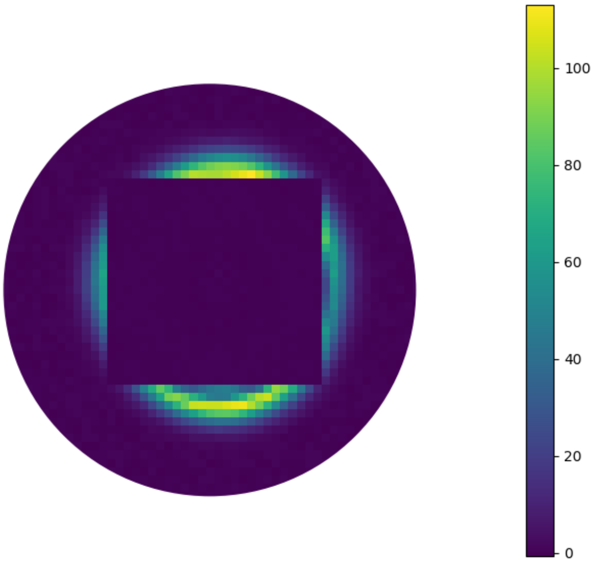


Figure 4.7: Point-to-point Flux difference between the Truth Subarray and the Imagette Linearised Subarray, for the aperture pixels. This simulation was made with no RON, with 15 stacked images and centroid variation σ of 1.5 pixels. The simulated star had a σ width of 5 pixels, in each axis.

Practical Example:

To better demonstrate this problem, let's describe and explore a practical example for a single pixel of the image:

1. For an exposure with 3-stacked images, where the individual truth values for a certain pixel are 10000, 2000 and 200, we will obtain a stacked truth flux of 12200.

2. When we add non-linearity errors, individually, to each of these 3 pixels, we will get non-corrected ADU values of 10159, 2066 and 212, respectively.
3. By stacking these three non-linear values we get a stacked flux of 12437.
4. Following the procedure of the uniform linearisation, we divide 12437 by the number of stacked images, 3, in this case, getting a value of around 4146.
5. We linearise this divided value, obtaining approximately 4054 ADU, and then multiply it by 3. We will obtain a corrected ADU value of 12161 ADU.

This value has a 39 ADU positive difference to the truth flux. Similar differences will occur in some of the aperture pixels where we have big variations from image-to-image, adding to the final error result, as can be seen in the pixel-to-pixel differences in figure 4.6, sub-figure **d**). Most of these differences are positive when we compare both fluxes, but in some situations can be slightly negative.

This difference is related to the CCD's correction curve: since it is non-linear, values of different regions of the curve are not linearly proportional to each other. For example, correcting a value of 600 will not be the same as correcting a value of 200 and then multiplying the result by 3. In different parts of the curve, we may have regions with a higher or lower slope, so when we uniformise the pixel values, we may go to a region of the curve where corrected ADU values will be approximated by defect/excess, depending on the curve's shape.

Performance Results:

We estimated the real flux computed with a circular aperture centered on each individual image. Then, we stacked these individual fluxes to have the sub-array's real flux.

Afterwards, we computed the flux of each linearised image using a centroid-centered circular aperture - for an exposure, the centroid will be the mean of all the individual image's centroids. These fluxes will then be compared to the real flux previously computed.

In figure 4.8 we present the reader with our performance results, done by simulating 15 exposures of 15 stacked images each. In the right sub-figure we present the results of a simulation without centroid variation and in the left sub-figure with a centroid's σ of 2.5 pixels. We represent our results in the form of a flux ratio between the linearised flux and the real flux, shown in the y-axis. In the x-axis it is represented the exposure number.

In the right sub-figure, we can observe that both linearisations give similar results when there is no centroid variation, as expected. The flux ratios can be over or below the control line, the cause being the RON.

On the other hand, in the left sub-figure, when using the imagettes, the flux ratios follow the control line more closely compared to the uniform linearisation method. It is important to note that the flux ratios in both situations are always below the control line. This is more noticeable in the uniform linearisation results, as the whole aperture pixels were not linearised in a way to account for image-to-image variations. As a consequence, the results for the linearisation correction method using the imagettes are also slightly below the control line, since we linearised some of the aperture pixels with the uniform linearisation method, as previously explained. The mean flux ratio for the imagette method is 0.999109 and for the uniform method is 0.990275. This means that if the truth value is one million ADUs, then the imagette method will fail to account 891 ADUs and the uniform method 9725 ADUs - an amount more than 10 times larger.

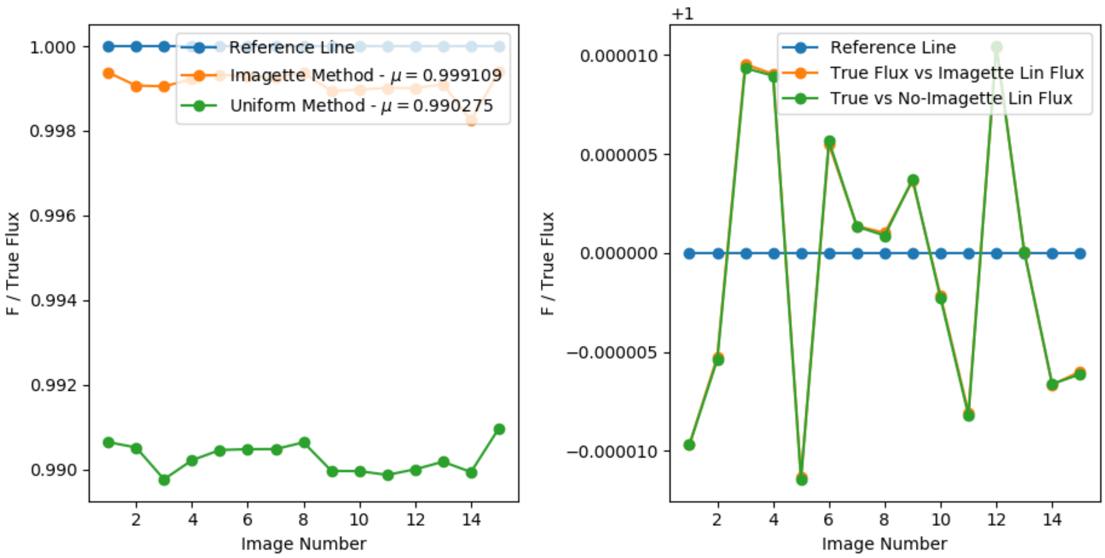


Figure 4.8: Linearisation Test Module final results: In blue we have the control line, in orange the imagette linearisation flux ratio and in green the uniform linearisation flux ratio. In the top-right corner we show the mean flux ratio for both situations. **Left:** Simulation made with all three errors present and a centroid σ of 2.5 pixels. **Right:** No centroid variation. Both simulations were made with **15** stacked images.

In figure 4.9 we present the relative photometric results for a simulation with a varying number of stacked images. Our goal was to assess the behaviour of each linearisation method when we stacked a higher or lower number of images. As a precision estimator, we used the P2P. The P2P was calculated in the same way as in the original DRP, explained previously. We can see from this figure, that in terms of point-to-point precision we generally get higher precisions (lower P2P) when using the imagette linearisation method, as the P2P is always lower than in the original uniform correction method.

Since during the mission the centroid is expected to change, we will get better results by using the imagette linearisation function.

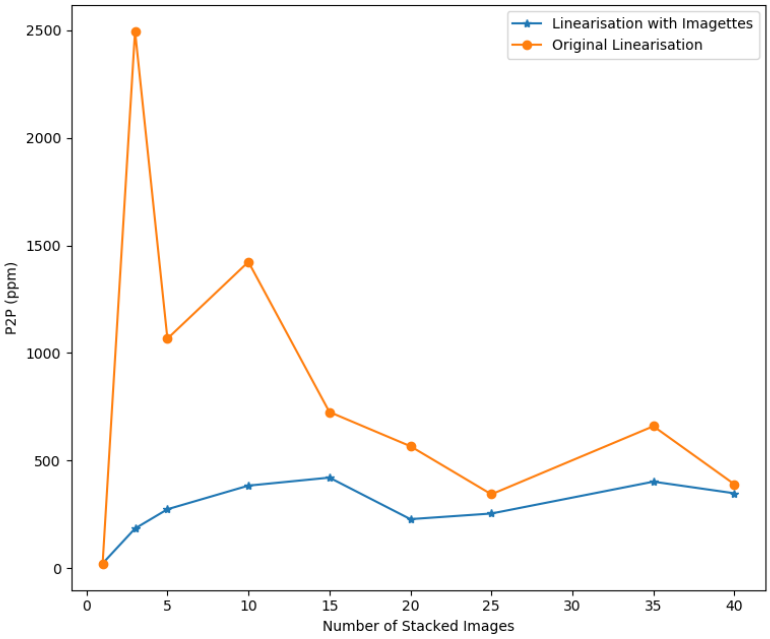


Figure 4.9: Linearisation Test Module final results- Simulation with a variable number of stacked images. On the y axis we represent the P2P in ppm. The RON was of 5 ADU rms and we used 25 exposures, with a centroid variation of 2.5 pixels and a star's maximum Amplitude of 64000 ADUs. In orange we present the results given by the original linearisation correction method and in blue the linearisation with imagettes method.

4.3 Offset Curve vs Low ADU Correction Curve

In this next subsection we present the results of additional simulations that were performed using the test module which was created for the previous subsection.

Due to the issues which would arise with the use of the splines curve, more specifically with the zero-offset of this curve, additional testing was performed in order to study the difference, performance-wise, between having a correction curve where we subtracted the offset to all the curve (the option adopted by CHEOPSim), so that it would start from the (0,0) point, or a curve which would only adapt the values belonging to the curve's first spline, so that it would also start from the origin. A representation of each of these curves is present in figure 4.10. In this figure we show the three distinct correction curves and we highlight the behaviour of each one of these for low read ADU values. The three curves present are:

- The original correction curve (**blue** in figure 4.10): this curve begins with an offset at (0,≈ 165). It is obtained from the original correction method.
- The offset-subtracted curve (**green**): we subtracted the zero-level offset of the previous curve to the whole curve, so that it will begin from the origin point.
- Low ADU-adjusted curve (**orange**): we adapted the lower ADU values, belonging to the first spline, so that the curve will start from the origin without having to offset the whole curve. The corrected ADU values for the first spline are calculated from a simple 1^{st} order linear function.

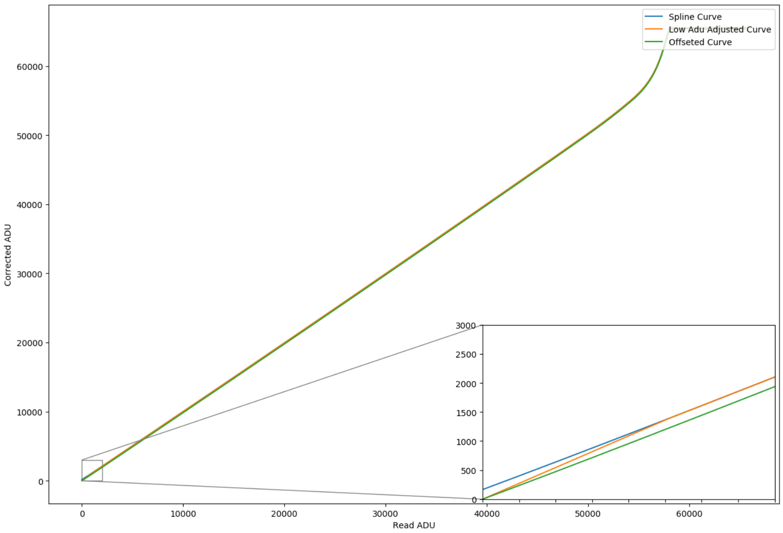


Figure 4.10: The three different correction curves: **Blue:** Spline Correction Curve. **Orange:** Low ADU-Adjusted Curve. **Green:** Offset Corrected Curve. In the bottom-right of the figure we can observe an augmented region of the correction curves for lower ADU values.

So that we could perform these new tests we had to slightly alter the code computed in the previous sections, so that the new curves could be accommodated in the new simulations.

Goals:

Our goal in these simulations was to understand how the two last curves would behave performance-wise, when they were used for correcting the non-linearised data, after non-linearity had been added using the splines curve.

We wanted to evaluate the performance of curves which would not have a bias offset present, since by using a curve with an offset would create situations where the absolute photometric difference between the calculated and true flux would highly increase. Finally, we wished to analyse the precision of using each curve, by analysing the relative photometric results.

Method:

In order to perform these tests, we simulated a star using the steps explained previously in this work (linearisation functional test subsection), except we added the non-linearity with the original correction curve and then afterwards corrected the non-linearised data with one of the two other curves (either offset or low-adu adjusted). In the end, we analysed the relative photometric flux results, which were produced when using each of the two different curves for non-linearity correction. In the previous sections we explained how the use of imagettes would provide us with better absolute and relative photometric results, therefore they were also used in these new simulations.

We made different simulations where we varied some of the distinct parameters simulation-to-simulation, more specifically, we varied the following parameters:

- Number of Exposures (between 3 and 20 images)
- Number of Stacked Images (between 3 and 20 stacked images)
- Centroid's Sigma (between 0 and 6 pixels)
- Simulated Star's Maximum Amplitude (between 10000 and 64000 ADUs)

Therefore, we varied each of these parameters individually while keeping the others static. As was previously done, we used the P2P of the light-curve to quantify its quality.

Results:

In figure 4.11, we present the results for the aforementioned two different curves:

In **a)** we present the results for a simulation where the number of exposures was changed, keeping the other parameters static, and the flux deviation for each case computed. In **b)** the number of stacked images was changing. In **c)** the centroid's sigma was varied. Conversely, in **d)** the star's maximum amplitude was varied.

In **a)**, we can observe that the option who produced the least overall deviation was the simulation using the offset curve (blue curve). For a higher number of exposures we reached precisions of 300 ppm using this curve. Additionally, in **b)** this option also provided better overall results than the low-ADU adjusted curve, where the precision was overall smaller for a changing number of stacked images.

In **c)** we can observe that both curves presented similar results for a varying centroid's deviation. It can also be seen that the flux deviation for both curves starts to highly increase for a centroid's sigma >3. This occurs because, for higher sigma values, the centroid will greatly change from unstacked-to-unstacked image, and the shown results should no longer be considered viable. Since the centroid is expected to only change for a small number of pixels, very high pixel variations should be discarded.

At last, in **d)** the offset curve produced the best overall results for a changing star's amplitude. This method was the one who overall showed higher flux precisions for star's with either a lower or higher maximum amplitude.

Discussion:

Overall, the curve which presented the best relative photometric results was the offset curve. For all the simulations done, this curve presented the highest precisions flux-wise. We can also note that, from the results presented in figure 4.11, the P2P when changing the centroid was the highest in all four parameter tests, being at least one order of magnitude higher than when the other three parameters were varied. We can yet again conclude, therefore, that the centroid's variation is extremely important to take into consideration when correcting the data in the CHEOPS pipeline.

Unfortunately, I did not have enough time to study the impact of using each of these correction curves in the transit depth for a real observational situation.

We are expecting the final calibration measurements on the instrument which will give us the final non-linearization curve. We expect that the problem with the point 0 will be fixed.

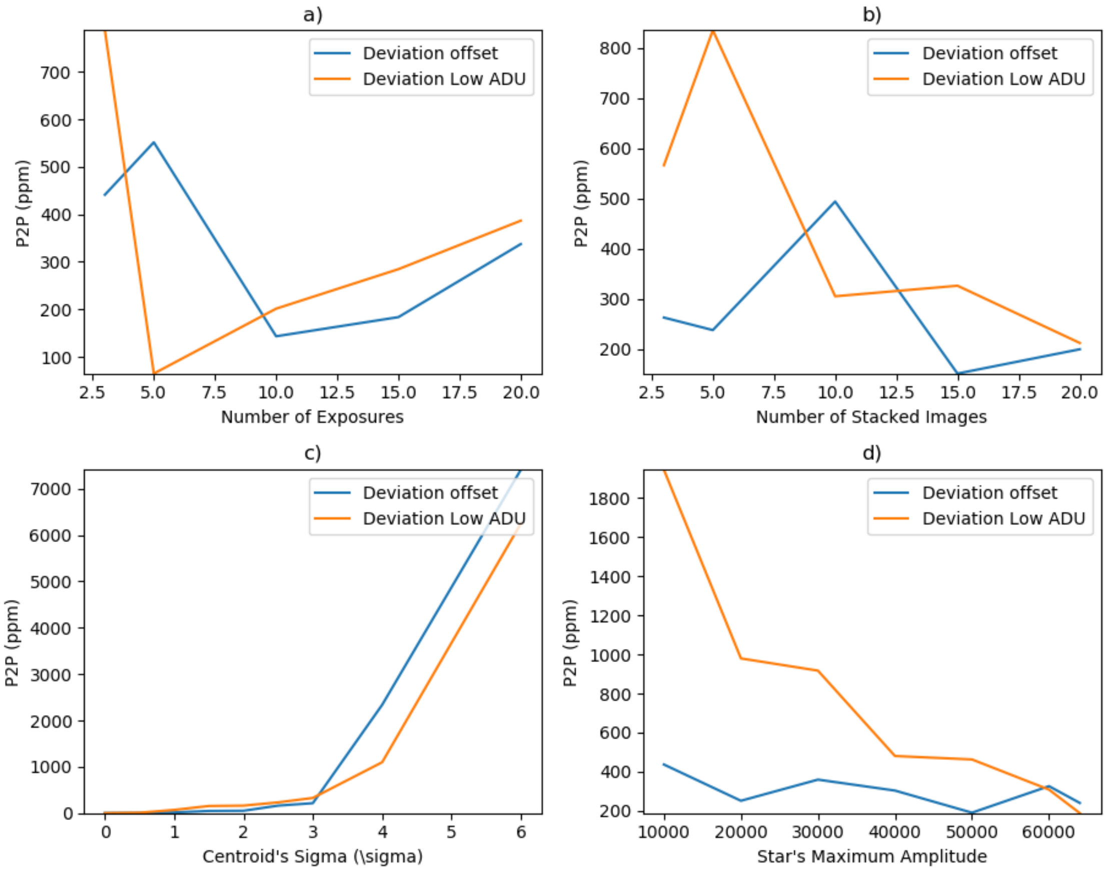


Figure 4.11: Correction Curve Simulation Results - in the y-axis of each sub-figure we show the P2P in ppm. The nominal values for the number of exposures,number of stacked images, centroid sigma and star's amplitude was 15 images, 15 stacked images, 2.5 pixels and 64000 ADUs, respectively. The blue curve is a simulation with the offseted curve and the orange with the low-adjusted curve. **a):** Varying number of exposures. **b):** Varying number of stacked images. **c):** Changing Centroid's sigma. **d):** Varying Star Maximum Amplitude.

4.4 Fractional Imagette Pixels

In the next subsection we present the results of the last case study that was developed in order to test the linearisation sub-module. We wanted to assess whether using the imagettes to correct certain pixel values, for which we would not have the full imagette information, would imrove the correction in comparison with the previous solutions presented. That is, back in subsection 4.1 and figure 4.1, we explained that, to correct non-linearity effects with the imagettes, it was necessary to determine the overlapping pixels of all the imagettes of an exposure, and, if a certain pixel would not be present in all of the exposure's imagettes we would correct it with the uniform method. Since by doing this we are discarding a lot of valuable information, we wanted to create a method that would maximize the use of all the imagette's information.

To do so, we established a fractional tolerance (*Input Fraction*) that will determine whether we will use the partial imagette information we have for a certain pixel. This *Input Fraction* is a variable that is inputed by the user, which determines if we use more or less partial imagette information. If a certain pixel is featured in a minimum number of imagettes - $N_{\text{imagettes}}$, so that its *Pixel Fraction* is equal or higher than the *Input Fraction*, we will accept it to be partially linearised:

$$\text{Pixel Fraction} = \frac{N_{\text{imagettes}}}{N_{\text{total}}} \geqslant \text{Input Fraction} \tag{4.1}$$

Pixel Fraction is thus the fraction between the number of imagettes where a certain pixel appears and the total number of imagettes for that exposure - N_{total}.

We have thus, three different sets of pixels:

- Where $0 \leqslant$ Pixel Fraction $<$ Input Fraction. These pixels will get linearised using the uniform method.
- Where Input Fraction $\leqslant$ Pixel Fraction < 1. These pixels appear in a fraction of imagettes equal or higher than the inputed fraction by the user. They will get linearised partially using information contained in the imagettes and partially using the sub-array, explained in the next subsection.
- Where Pixel Fraction $= 1$. These are the pixels for which we have the full imagette information, therefore we linearise them with the method presented back in subsection 4.1.

Method:

For the second case, we compute the individual missing pixel value, of exposure *exp* and position *i,j*, as:

$$\text{Individual Pixel Value}_{exp,i,j} = \frac{(\text{Unbias Sub}_{exp,i,j} - \Sigma\text{Imagette Unbias}_{exp,i,j}) * \delta_{exp,i,j}}{N_{stacked} * (1 - \text{Pixel Fraction}_{exp,i,j})} \tag{4.2}$$

In this equation, Unbias $\text{Sub}_{exp,i,j}$ is the stacked unbias sub-array value for that pixel, ΣImagette $\text{Unbias}_{exp,i,j}$ is the sum of the known unbias imagette values of that pixel and $\delta_{exp,i,j}$ takes the value of 1 if that pixel satisfies the second condition or 0 if otherwise. We divide the numerator by $N_{stacked} * (1 - \text{Pixel Fraction}_{exp,i,j})$ to have the mean unknown value of the individual pixel.

Afterwards, we will linearise this mean value, and then multiply it by the number of imagettes where it does not appear, so to have the missing pixel information. At last, we will add to this value the sum of the linearised Imagette, $\Sigma\text{Linearised Imagette}_{exp,i,j}$.

Practical Example:

As was done previously, in order to better explain the process, we use a practical example of an exposure with 5 stacked images, if we consider a pixel p that features in 3 of the 5 imagettes (so Pixel Fraction = 0.6) and we determine an Input Fraction of 0.5, the second case is fulfilled and that pixel will be fractionally linearised. We will thus compute its mean individual value using equation 4.3. Afterwards, we linearise it and then multiply it by 2, since it was not present in 2 of the 5 imagettes. Finally, the final sub-array value for this pixel will be equal to the sum of the 3 linearised pixels, present in the imagettes, plus twice (2 of 5 missing) the individual linearised value (Individual Pixel Value in equation 4.3), computed using the previously referred procedure.

In this way, if a certain pixel is featured in a high number of imagettes, even if not all, we may have better overall flux results by using this partial information than by fully linearising that pixel with the uniform method.

Results:

We tested this new method by making different simulations where we changed some of the inputed variables (centroid sigma, star's maximum amplitude and number of stacked images) and varied the Input Fraction. In this way, we tested whether by using more or less partial information would improve (or not) the absolute photometric results, in different situations. The testing setup was the same as in subsection 4.1, where we simulated a star, added non-linearity using an interpolated offset curve, added a Bias/RON error and corrected the data afterwards, using the offseted curve.

We simulated single exposures, with a standard value of 30 stacked images, with a Bias and RON of 2600 ADUs and 5 ADUs rms, respectively. The nominal value for the centroid's sigma and the star's maximum amplitude were 2.5 pixels and 64000 ADUs, respectively.

We present the obtained results in figure 4.12. In the y-axis of this figure, we present the flux ratio (= Linearised Flux/Truth Flux) and in the x-axis the *Input Fraction*:

- In the top sub-figure, we varied the centroid's sigma and kept the star's maximum Amplitude at 64000 and the number of stacked images at 30. We can note that for a centroid sigmas $\geqslant 1.5$ we start noting a slight tendency for the flux ratios to lower when we use less partial imagette information (higher input fraction). This effect is more notorious for higher variations of centroid sigma - 2.5 and 3 for example.
- In the middle sub-figure, we changed the star's maximum amplitude and kept a static centroid sigma of 2.5 and 30 stacked images. We can note more clearly the same tendency as in the previous sub-figure.
- In the bottom sub-figure, we varied the number of stacked images and kept a centroid sigma of 2.5 and a star's maximum Amplitude of 64000. In this case we can also clearly note that when we use less information we have worst flux results than the other way around.

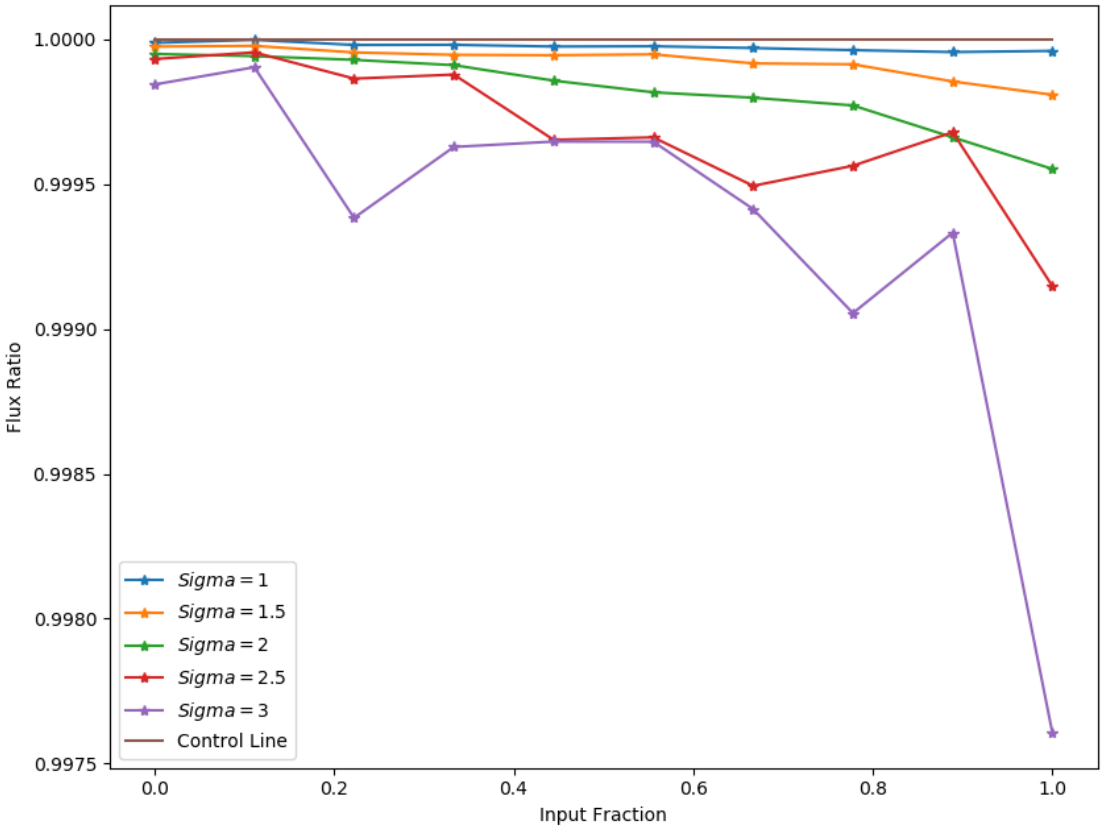
In addition, we present the relative photometric results for a simulation with a variable number of stacked images and nominal values of 25 exposures, centroid's sigma of 2.5 pixels, RON of 5 ADU rms and star's maximum amplitude of 64000 ADUs. The results are shown in figure 4.13. We can see that using more information will provide us with higher precisions. In all simulations, the precision of the method with fractional imagettes was always lower than 100 ppm, whereas the normal imagette method was less precise.

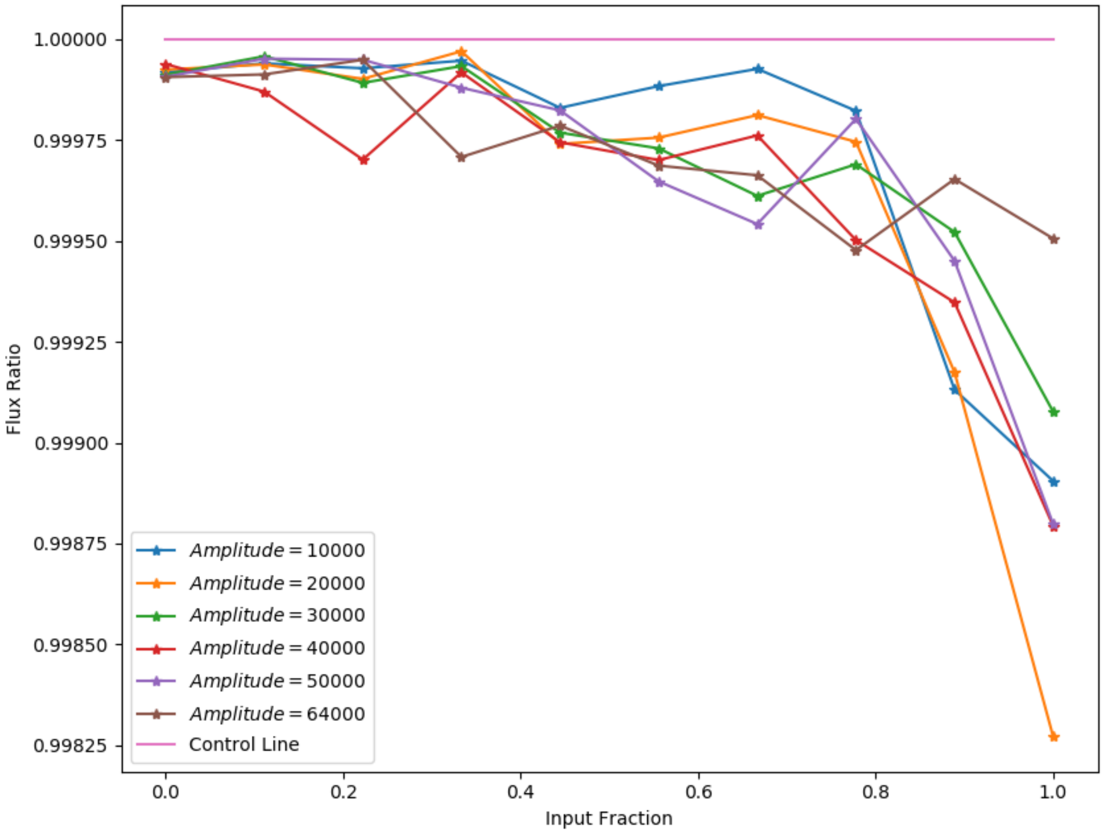
Discussion:

We can conclude from the results that, as expected, we will generally obtain a linearised flux closer to the real one when using all the available information possible. Albeit with some fluctuations, we can see a clear downtrend when using less imagette information.

These results are strengthened by analysing the relative photometric results comparing the normal vs fractional imagette method. In all simulations the latter was more precise by a considerable margin.

Therefore this new method of using also the fractional pixels for which we don't have the full information will provide us with both better absolute and relative photometric results than when using only the full-information pixels.


Flux Ratio
1.0000
0.9995
0.9990
0.9985
0.9980
0.9975
Input Fraction
0.0
0.2
0.4
0.6
0.8
1.0
$Sigma = 1$
$Sigma = 1.5$
$Sigma = 2$
$Sigma = 2.5$
$Sigma = 3$
Control Line


Flux Ratio
1.00000
0.99975
0.99950
0.99925
0.99900
0.99875
0.99850
0.99825
Input Fraction
0.0
0.2
0.4
0.6
0.8
1.0
$Amplitude = 10000$
$Amplitude = 20000$
$Amplitude = 30000$
$Amplitude = 40000$
$Amplitude = 50000$
$Amplitude = 64000$
Control Line

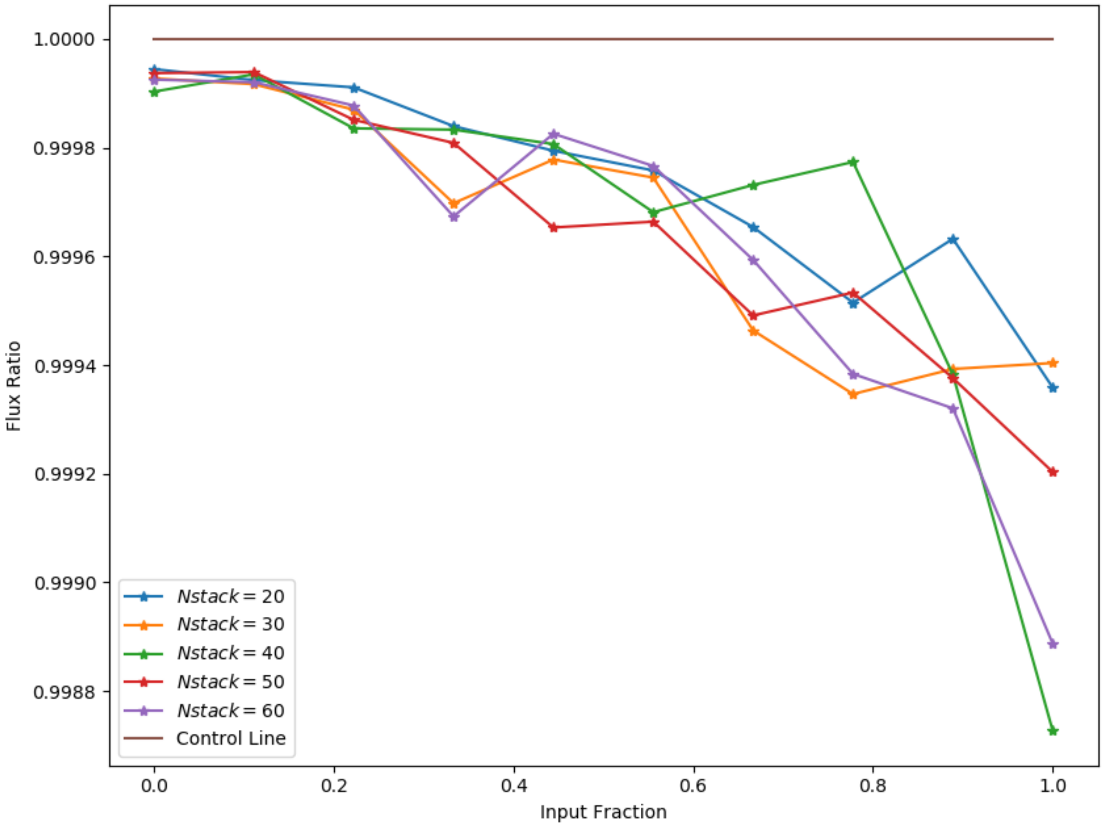


Figure 4.12: Fractional Imagette Pixels Results. In the y-axis it is represented the Flux Ratio between the measured linearised flux and the truth flux and in the x-axis the Input Fraction. **Top:** We varied the centroid, keeping the other variables static. **Middle:** We changed the simulated star's maximum amplitude. **Bottom:** We varied the number of stacked images.

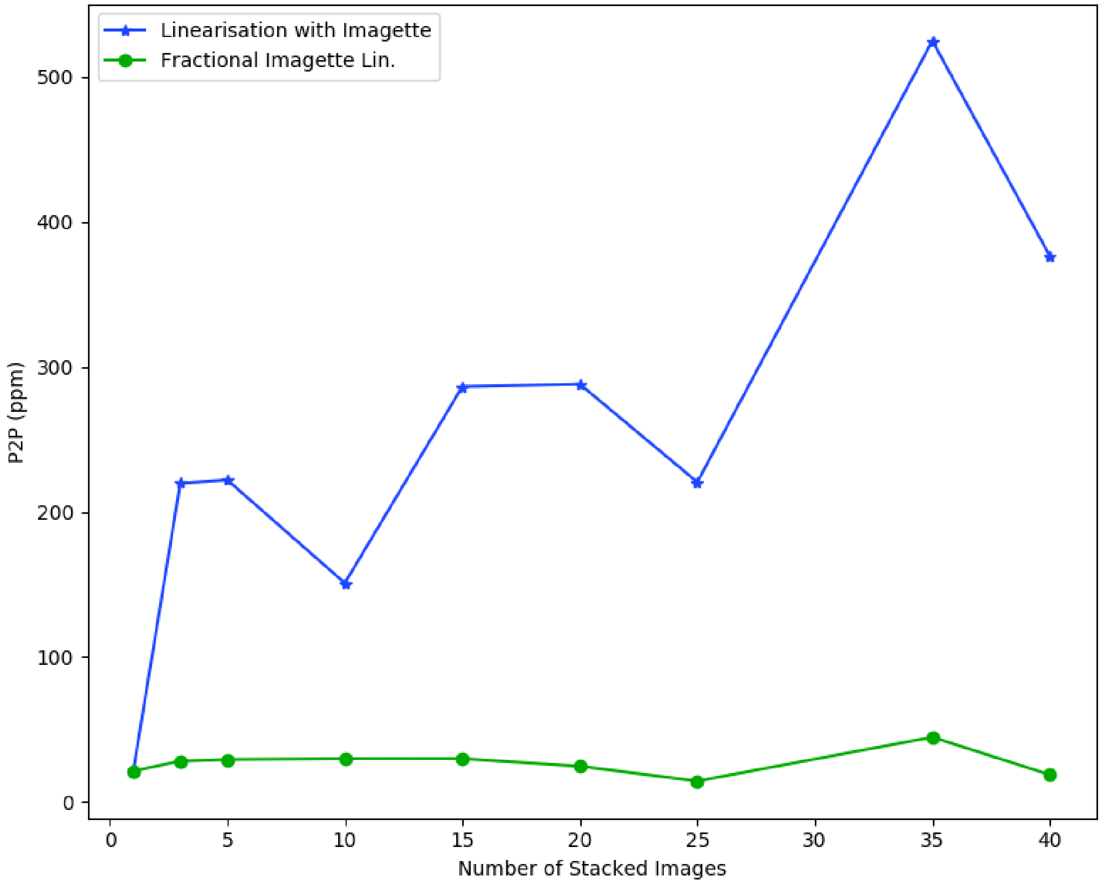


Figure 4.13: Fractional Imagette Relative Photometric Results. In the y-axis it is represented the P2P in ppm. The simulation was made with a variable number of stacked images and with nominal values of 25 exposures, centroid's sigma of 2.5 pixels, RON of 5 ADU rms and star's maximum amplitude of 64000 ADUs. In blue we represent the imagette method, without the use of fractional pixels (*Input Fraction = 1*). In green we present the results for an imagette method that uses all the available information (*Input Fraction = 0*).

5

Conclusions

This work set to study and improve the CHEOPS data reduction pipeline, being developed by collaboration between two Astrophysics Department (IA and LAM). The sub-module chosen for improvements was the linearisation correction. It also set to replicate the original pipeline for easiness of additional development and testing.

We presented a replicated data reduction pipeline with a new improved sub-module.

We showed in this work that the new linearisation sub-module could improve the overall results of the pipeline, and that it would highly increase both the absolute and relative photometric measurements that will later be performed by CHEOPS. At first, we used a method which would only use the imagettes for the pixels where we would have the full information and showed that this would provide better results than the uniform method that was already used in the CHEOPS pipeline.

Also, due to problems arising when treating low ADU values (offset problem of the original curve) we tried to evaluate the output of using two different correction curves, adapted differently to low ADU values. We concluded that it was yet unclear how the CCD will behave for lower read values, therefore there should later be an adaptation of the linearisation curves depending on this output. Nevertheless, the offseted curve provided us with better relative photometric results, for most simulations.

At last, we took a step further and tested whether by using the maximum available information stored in the imagettes, we would obtain better performance results than using the first tested imagette method. From this test we concluded that by using more information, no matter how little, would overall provide us with better absolute photometric results. In addition, we tested the relative photometric results for the cases where we would use no fractional information and compared them to the cases where we would use all the available information. We concluded that the latter method is more precise than the former.

We conclude then that the imagette method should be used in the original pipeline, as it will provide better results compared to the original uniform correction method.

6

Bibliography

S. Aigrain, S. T. Hodgkin, M. J. Irwin, J. R. Lewis, and S. J. Roberts. Precise time serie photometry for the Kepler-2.0 mission. *Monthly Notices of the Royal Astronomical Society*, 447:2880–2893, March 2015.

M. Auvergne et al. The CoRoT satellite in flight: description and performance. *Astronomy and Astrophysics*, 506:411–424, October 2009. doi: 10.1051/0004-6361/200810860.

P. Bordé et al. Transiting exoplanets from the CoRoT space mission. XI. CoRoT-8b: a hot and dense sub-Saturn around a K1 dwarf. *Astronomy and Astrophysics*, 520:A66, September 2010. doi: 10.1051/0004-6361/201014775.

S. T. Bryson et al. The Kepler Pixel Response Function. *Astrophysical Journal, Letters*, 713:L97–L102, April 2010. doi: 10.1088/2041-8205/713/2/L97.

D. Charbonneau, T. M. Brown, D. W. Latham, and M. Mayor. Detection of Planetary Transits Across a Sun-like Star. *Astrophysical Journal, Letters*, 529:L45–L48, January 2000. doi: 10.1086/312457.

D. J. Christian et al. The SuperWASP wide-field exoplanetary transit survey: candidates from fields 23 h RA 03 h. *Monthly Notices of the Royal Astronomical Society*, 372:1117–1128, November 2006. doi: 10.1111/j.1365-2966.2006.10913.x.

J. L. Christiansen et al. The Derivation, Properties, and Value of Keplers Combined Differential Photometric Precision. *Publications of the Astronomical Society of the Pacific*, 124:1279, December 2012. doi: 10.1086/668847.

A. Deline, M. Sordet, F. Wildi, and B. Chazelas. The testing and characterization of the CHEOPS CCDs. In *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, volume 10562 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 105624E, September 2017. doi: 10.1117/12.2296042.

R. Drummond, V. Lapeyrere, M. Auvergne, B. Vandenbussche, C. Aerts, R. Samadi, and J. E. S. Costa. Correcting for background changes in CoRoT exoplanet data. , 487:1209–1220, September 2008. doi: 10.1051/0004-6361:200809639.

e2v technologies (uk) limited. *CCD47–20 Back - Datasheet*, 2006.

A. Fortier, B. Thomas, B. Willy, B. Christopher, C. Virginie, D. Ehrenreich, and T. Nicolas. Cheops: a space telescope for ultra-high precision photometry of exoplanet transits. *Proc.SPIE*, 9143:9143 – 9143 – 12, 2014. doi: 10.1117/12.2056687. URL https://doi.org/10.1117/12.2056687.

E. Gaidos, D. Kitzmann, and K. Heng. Exoplanet characterization by multi-observatory transit photometry with TESS and CHEOPS. *Monthly Notices of the Royal Astronomical Society*, 468:3418–3427, July 2017. doi: 10.1093/mnras/stx615.

M. Gillon et al. Seven temperate terrestrial planets around the nearby ultracool dwarf star TRAPPIST-1. *Nature*, 542:456–460, February 2017. doi: 10.1038/nature21360.

CHEOPS ground calibration team. *CHEOPS-UGE-SYS-PR-019 2.2 data reduction procedures*.

D. Hoaglin, F. Mosteller, and J.W. Tukey. Understanding robust and exploratory data analysis. 1983.

S. B. Howell. *Handbook of CCD Astronomy*. March 2006.

S. Hoyer. Cheops data reduction pipeline. In *CHEOPS Fifth Science Workshop-Schloss Seggau, Austria*, jul 2017.

J.R. Janesick. *Scientific Charge-coupled Devices*. Press Monograph Series. Society of Photo Optical, 2001. ISBN 9780819436986. URL https://books.google.pt/books?id=rkgBkbDie7kC.

J. M. Jenkins et al. Overview of the Kepler Science Processing Pipeline. *Astrophysical Journal, Letters*, 713:L87–L91, April 2010. doi: 10.1088/2041-8205/713/2/L87.

M. Mayor and D. Queloz. A Jupiter-mass companion to a solar-type star. *Nature*, 378:355–359, November 1995. doi: 10.1038/378355a0.

M. Perryman. *The Exoplanet Handbook*. January 2014.

E. V. Quintana, J. M. Jenkins, B. D. Clarke, H. Chandrasekaran, J. D. Twicken, S. D. McCauliff, M. T. Cote, T. C. Klaus, C. Allen, D. A. Caldwell, and S. T. Bryson. Pixel-level calibration in the Kepler Science Operations Center pipeline. In *Software and Cyberinfrastructure for Astronomy*, volume 7740 of , page 77401X, July 2010. doi: 10.1117/12.857678.

ESA Redbook. *CHEOPS Definition Study Report (Redbook)*, nov 2013.

R. Samadi, F. Fialho, J. E. S. Costa, R. Drummond, L. Pinheiro Da Silva, F. Baudin, P. Boumier, and L. Jorda. The Corot Book: Chap. V.5/ Extraction of the photometric information : corrections. *ArXiv Astrophysics e-prints*, March 2007.

S. Tavernier. *Experimental Techniques in Nuclear and Particle Physics.* 2000.

P. G. van Dokkum. Cosmic-Ray Rejection by Laplacian Edge Detection. *Publications of the Astronomical Society of the Pacific*, 113:1420–1427, November 2001. doi: 10.1086/323894.

A. Vyas, M. Roopashree, and B. Prasad. Optimization of Existing Centroiding Algorithms for Shack Hartmann Sensor. *ArXiv e-prints*, August 2009.

WCS. https://fits.gsfc.nasa.gov/fits$_w$*cs.html.*

P. J. Wheatley et al. The Next Generation Transit Survey (NGTS). *Monthly Notices of the Royal Astronomical Society*, 475:4476–4493, April 2018. doi: 10.1093/mnras/stx2836.

Appendices

A

Pipeline Manual

A.1 File Formats and Packages

Throughout the work there were used and handled different types of data-formats, from which we can point out the following:

- Data FITS files (.fits): FITS is a digital file format that contains an header, where we store general information about the data, such as time of observation, the data formats, etc, and it also contains a data part, where information is stored in tables. Each of the table's columns may have a different format than the other, so that we can store different types of information.

- Python scripting files (.py): These are the algorithms files of the pipeline, each file corresponds to a different sub-module.

- Config file (.yml): contains configuration paths and options, so that different people can easily adapt and run the pipeline. This file can be changed at will, depending on the developer's own PATHS and configuration. More specifically, it contains a data-set path, a path to the reference files and an output data path, to store the output data of the DRP.

There were also used different python libraries (packages) that served distinct purposes. We can note:

- Numpy - allows to easily create and manipulate arrays of data

- Scipy - provides us with mathematical expressions and calculations

- Matplotlib - For data visualization and representation

- Astropy - for reading/storing data from/in FITS files as well as coordinate conversion.

A.2 Replicated Pipeline Overview

A replicated pipeline was developed in order to simplify the development and testing of new algorithms. This new version ran the same modules as the original one but was simpler, since it did not have external interfaces or coding constraints. We uploaded this new version to a private repository on Github, for version control purposes and continuous development. We created a single python scripting file (.py) as the main pipeline file. This file is the core of the replicated pipeline and it initializes all the modules sequentially, therefore, if a new sub-module needs to be tested, we can just integrate it with this core file. We tried to keep the modular structure of the original pipeline as much as possible, therefore, we kept the division of each sub-module inside each of the four modules: Calibration, Correction, Photometric Extraction and Dr-Tools.

The data-sets and the reference files, as previously mentioned, are stored inside FITS files.

Finally, in order to allow different people to be able to easily download and run the code, we included a config file (.yml). The main file will then fetch the information contained in this file, in order to correctly access the data-sets and reference files.

Each time the pipeline is run, the output data will be stored inside a FITS file, which will contain the most important information, from the output light-curve to the different error sources, etc. We will also directly plot some of this information for the user to analyse, as represented in figure A.1.

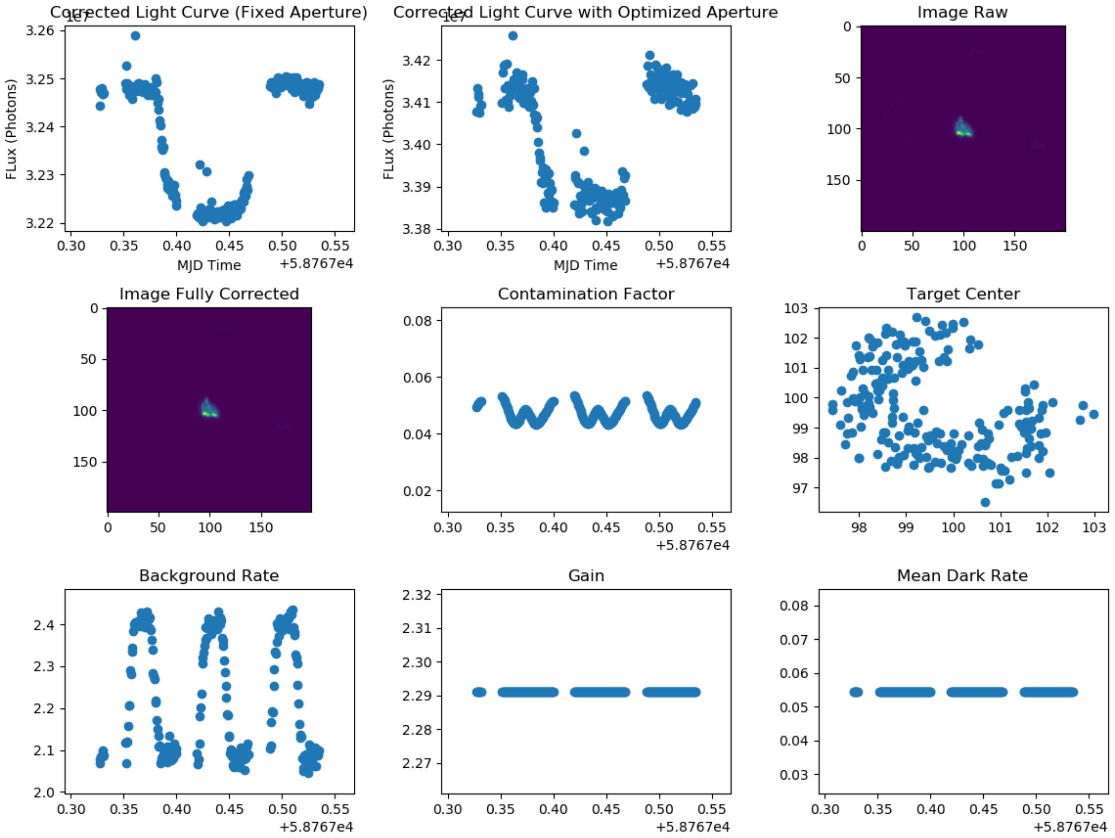


Figure A.1: Pipeline Final Plot Example. We show the light curve, contamination, the RAW and corrected sub-array, the target's center variation, the background and dark rate and the gain.